



APRIL 1987

# QUASAR

- |     |   |     |  |
|-----|---|-----|--|
| 465 | van de redactie...  | 477 | Microsoft op de QL. Een basic die 'vreemd gaat'.           |
| 466 | Ard maak GRRReluIden. enkele kanttekeningen bij programma's.            | 478 | Archivist. Software bespreking.                            |
| 466 | Opmerkingen over SuperBasic;Ard   | 479 | Toevoegen van velden in Archive. Procs....                 |
| 467 | Errata alternatief fount Quill. (Iedereen maakt wel eens vergissing...) | 481 | QL-tolk. QL en Atari in de bocht.                          |
| 468 | Ingezonden mededelingen; te koop, gevraagd e.d.                         | 483 | Prisoner. Het dilemma nader bekeken.                       |
| 468 | Tandy plotter CGP115. Veel Interesse...                                 | 484 | Gelezen.QLTurboQuill, Turbo SB Compiler, Care electronics. |
| 469 | Prof.Astrologer; een softwarebespreking.                                | 485 | Selko datamanager. ....                                    |
| 470 | 'Kleintje' datum, pl en 13. Enkele geconverteerde programmaatjes.       | 486 | Over Databases en Archive. Bob komt over de brug.          |
| 471 | 'Vooruitblikje'   | 487 | Onderdelen Info.Sluitingsdata kopij, 500ste lld.           |
| 472 | Tandata modem. Een bespreking.  | 488 | Meteosat; Datum Bijeenkomst.                               |
| 473 | Contractor. Een SB-programma dat 'krimpt' in de was.                    | 489 | CURSUS MACHINETAAL DEEL 1.                                 |
| 475 | All Play All Competitie. Een handige hulp bij het organiseren.(SB)      | 494 | Overzicht SIN_QL_AIR software.                             |
| 477 | FINDNAME_BAS, v 1.1. Een programma dat zich 'naar-naar' zoekt.          | 497 | Overzicht QUANTA software.                                 |
|     |   | 502 | De andere kant van QUASAR.                                 |







**Ledenadministratie, Sekretariaat,  
Penningmeester, Nabestellingen oude  
nummers:**

**BOB VISSER, Scheepmakerskade 30,  
3011 VX Rotterdam. 010 - 414.3554**



**Redaktie, Layout en samenstelling  
Quasar:**

**RON DEN BREEMS, Kroonstaddreef 27,  
3067 RT Rotterdam. 010 - 455.1234**



**Cartridge/Disk-service,**

**Vragentelefoon voor  
SuperBasic/Pascal/Hardware:**

**KEES VAN DER WAL,  
20.30 - 23.30 uur: 015 - 120.360**



**Hardware en Onderdelen, Reparaties:**

**MICHEL & WIM SPANJER,  
19.00 - 20.00 uur: 03404 - 20581**



**Machinetaal en hardware  
vragentelefoon:**

**ARD JONKER, 020 - 230.795**



**C en machinetaal vragentelefoon:**

**MARC KOOL, 020 - 429.345**



**Vertalers:**

**FRED BRANDS, Bruggemanlanden 25,  
7542 CA Enschede. 053 - 77.22.11**

**RUUD REDDINGIUS, Ruysdaelstraat 73b,  
1071 XB Amsterdam. 020 - 643.544**



**Vooraf voor nieuwe leden nog even het volgende:**

**SIN\_QL\_AIR is een stichting. Voor deze vorm is  
gekozen om het functioneren van de club, zoals  
wij hem toch altijd noemen, eenvoudig te houden  
en efficiënt. In de statuten van de stichting, die  
ieder lid aan kan vragen, of kan vinden in uitgave  
nummer 21 van Quasar, staat dat het doel van de  
stichting is:**

**Het bevorderen van het gebruik van en de  
uitwisseling van ervaringen, denkbeelden en  
programmatuur door gebruikers en belangstel-  
lenden voor de personal computer en de daarbij  
behorende of te verschijnen randapparatuur in het  
algemeen, en in het bijzonder voor de computer**

**van het merk Sinclair type QL en de daarbij beho-  
rende of te verschijnen randapparatuur.**

**Dan volgt er een stukje waarin staat hoe dat doel  
nagestreefd wordt:**

**a/ het minimaal zes maal per jaar uitbrengen van  
een nieuwsbulletin genaamd Quasar.**

**b/ het tegen betaling beschikbaar stellen van  
public domain software door middel van de  
cartridge-service.**

**c/ het organiseren van bijeenkomsten.**

**d/ het onderhouden van contacten met lever-  
anciers en zusterorganisaties in het buitenland.**

**e/ waar mogelijk het gezamenlijk inkopen van ver-  
bruiksmaterialen.**

**f/ het bieden van telefonische of schriftelijke hulp  
bij problemen.**

**Volgens dezelfde statuten heeft onze stichting  
minimaal twee bestuursleden; een voorzitter en  
een penningmeester/sekretaris. (Ron & Bob)**

**Sommige mensen staan wat huiverig tegenover  
het idee stichting, omdat een stichting geen  
ledenvergaderingen kent; men is dan bang geen  
inspraak te hebben. Het is juist de vrijheid om niet  
te hoeven vergaderen die ons ertoe gebracht  
heeft voor de stichtingsvorm te kiezen; bij een  
landelijke 'vereniging' is vergaderen erg lastig.**

**Het 'beleid' is er echter op gericht om zoveel  
mogelijk aan de wensen van alle 'leden' (die dus  
eigenlijk donateurs heten) tegemoet te komen.**

**De belangrijkste functie van sin\_QL\_air is onge-  
twijfeld het uitbrengen van een nieuwsbrief; de  
inhoud daarvan bestaat altijd voor het allergrootste  
deel uit bijdragen van leden! Niet alleen dragen de  
leden bij aan de inhoud van de Quasar, ze dragen  
ook bij door het verzamelen van informatie, het  
maken van hardware, het vinden van lage prijzen,  
het organiseren van bijeenkomsten, het telefo-  
nisch beantwoorden van vragen enz. enz.**

**SIN\_QL\_AIR is dus VAN en VOOR u!**

**Een lidmaatschapjaar loopt van april tot april.**

**Voor f40,- krijgt u een jaar lang de Quasar in de  
bus en kunt u gebruik maken van de faciliteiten die  
sin\_QL\_air te bieden heeft.**

**ZIE VOOR**

**GOEDKOPE CARTRIDGES**

**DE MEDEDELINGEN VAN**

**DE  
CARTRIDGE/DISK-SERVICE !!**





### BEDANKT JOHN!

Zo ongemerkt is John Smits geen redakteur meer maar ik wilde het niet nalaten hem hartelijk te bedanken voor het fantastische werk dat hij gedaan heeft. Zijn hand herkende we in de degelijke softwarebesprekingen en de fleurige layout. (Zoals ik al eens zei: illustreren is nog niet mijn sterke zijde; ik blijf hopen) John zal misschien in de toekomst nog wel bijdragen leveren in de vorm van artikelen. We kijken er naar uit John! Nogmaals bedankt.

### 25 en 2 maal 500...

Dit is alweer het 25ste nummer van Quasar; we zijn over de 500 pagina's en over de vijfhonderd leden. 524 om precies te zijn. Dat 289 daarvan op dit moment nog moeten betalen kan de feestvreugde niet minder maken.

Druk, drukker, .....

Het is zeer druk 'op de redactie' en ook Bob heeft zijn handen vol aan het bijhouden van alle betalingen, want er zijn natuurlijk zo langzamerhand alweer 524-289=235 betalingen te verwerken. (reken ik wel goed zonder computer?)

Ondertussen hebben we verschillende reacties gehad op onze oproep voor goedkoper drukwerk. De interessantste was van een lid van de club die een offsetdrukkerij heeft. Vanaf dit nummer dus drukwerk. Ook de verzending wordt door onze drukker gedaan. Toch bedank voor de andere reacties.

Veel, veel artikelen. Zo veel op dit moment, dat ik een aantal artikelen tot de volgende quasar bewaar omdat meer dan het huidige aantal pagina's te gek zou worden. Sommige kopij kwam ook te laat voor dit nummer binnen. Elders in dit nummer staan wat data die ik als streefdata wil houden voor volgende quasars. Dit alles neemt natuurlijk niet weg dat ik blijf roepen om kopij. Zeker gezien de verscheidenheid aan artikelen deze keer vertrouw ik erop dat zowel beginners als gevorderden hun Quill-peutjes of pennevruchten op blijven sturen.

### Voorproefjes.

Volgende keer een uitgebreid verhaal over microdrives; hoe zit zo'n cartridge in elkaar en hoe maakt de QL gebruik van de tape? Kees zet het haarfijn uiteen.

### Bijeenkomst.

Jawel alweer een bijeenkomst ook; op 23 mei in Utrecht. Zie elders in dit nummer de routebeschrijving.

### Prijs.

Het vijfhonderdste lid krijgt op deze bijeenkomst het Miracle modem aangeboden. Wie is dat 500ste lid? Zoek in dit nummer en u weet of u de gelukkige bent.

### "the beginners' elite"

Enkele beginners programma's deze keer en een super-procedure die zegeviert in ons prisoners' dilemma.

Enfin, er is teveel om op te noemen en ik ben te vermoeid om nog zinnige dingen te zeggen. Bob heeft zijn zegje gedaan in een artikel over database begrippen en Archive en zal me deze keer niet uit kunnen lachen om het kleine lettertje zoals ik dat vorige keer nodig had om nog al mijn r0daktionneel op één kantje te krijgen.

Hier laat ik het dus gewoon bij.  
Veel leesplezier en tot .....  
Ron.®

**QL bijeenkomst**  
zaterdag 23 mei  
Grebbeberglaan 15 Utrecht



## ARD maakt GRRReluiden

Trek het U niet aan, het is niet persoonlijk bedoeld. Op pagina 448 van de vorige Quasar heb ik me even flink geergerd. Niet dat iemand zich daar ook maar IETS van hoeft aan te trekken, maar liever niet meer zo programmeren. De schrijver heeft al gezegd dat kritiek welkom is, daar gaat-ie. Ik zal mijn best doen het opbouwend te houden. Als je een 'SElect ON' programmeert, wordt het een stuk duidelijker als je één of twee spaties inspringt, om aan te geven hoe diep je genest bent:

niet zo:	maar zo:
SElect ON a	SElect ON reutel
ON a=7:IF dit THEN dat	= 7:IF dit THEN dat
ON a=15:IF zus THEN zo	=15:IF zus THEN zo
END SElect	END SElect

Verder is het jammer dat het programma alleen van mdv werkt; tenmiste als je niet 11 keer 'mdv' verandert in 'lp', of 'hdk' of 'fdk' of wat\_dan\_ook. Het is handiger om aan het begin van het programma een variabele te vullen, die aangeeft waar het lekkers te halen valt; net zoals je gedaan hebt met drivea\$ en driveb\$. Inmiddels ben ik er achter dat alléén een 'dev\$' niet voldoende is. Nu gebruik ik er 2; één voor het data device en één voor het programma device.

```
prog_dev$='hdk1_'
data_dev$='hdk2_'
```

Dijkstra (een haast genieaal Informaticus die nederland heeft vernield voor het land der schier onbegrensde mogelijkheden) heeft al lang geleden een rapport geschreven onder de naam 'GOTO considered harmful' (GOTO beschouwd als schadelijk). Dat blijft onverkort gehandhaafd. GEEN GOTO dus!

Om uit een procedure te komen kan met 'RETurn' gebruiken, of EXIT.

Tenslotte nog een truuk om te bepalen hoeveel ruimte er in het array dir\$ gereserveerd moet worden: (regel 1080)

Open tussen regel 1120 en 1130 de file

```
data_dev$&drive$&'_dir'
```

en lepel hem leeg. Het aantal regels dat hieruit te halen valt voor er EOF optreedt is drie (of twee, dat weet ik niet precies) méér dan het aantal files dat er op disc staat.

```
1121 OPEN_IN#6,.....
1122 aantal_regels = 0:max_lengte=0
1123 REPEAT tel_regels
1124 IF EOF(#6) THEN EXIT tel_regels
1125 input #6,dummy$
1126 aantal_regels=aantal_regels+1
1127 IF LEN(dummy$)>max_lengte THEN
    max_lengte =LEN(dummy$)
1128 END REPEAT tel_regels
```

```
1129 DIM DIR$(aantal_regels,max_lengte)
```

Dit geeft ook al een hint wat betreft het slordige testen

op het einde van de file in regel 1160. Beter ware:

```
1145 IF EOF(#7) THEN EXIT c
```

en regel 1080, 1110, 1160 en 1190 laten vervallen. (wel even de oude file weggooien zoals dat in 1080 gebeurt.)

De test IF EOF(#...) kijkt of het gespecificeerde kanaal al

aan het einde van de file gekomen is.

ARD

### Opmerkingen over SuperBasic programmas

Nog vaak komen er programma's voor die, weliswaar netjes geschreven, verbeterd zouden kunnen worden. Zo zou ik graag even terug komen op het programma 'geta' op pagina 445 en verder. We zien in regel 120 tot regel 1210 een aantal malen dezelfde serie opdrachten. We kunnen dit vervangen door een procedure die 'kaart' heet. Het zou er zo uit kunnen zien:

```
DEFine PROCedure init
```

```
    LET getal = 0
```

```
    screen_def$='scr_200x200a40,40'
```

```
END DEFine init
```

```
:
```

```
:REMark hoofdprogramma
```

```
:
```

```
    FOR kaart = 0 TO 6
```

```
        LET bijdrage = 2 ^ kaart:REMark dit zijn
        namelijk
```

```
        vraag_kaart
```

```
    kaart,bijdrage
```

```
    END FOR kaart
```

```
    PRINT 'GROTE TRUUK ...etc.\
```

```
    PRINT 'uw kaart was',getal
```

```
:
```

```
DEFine PROCedure
```

```
    vraag_kaart(kaart_nummer,bijdrage)
```

```
    toon_kaart kaart_nummer
```

```
    IF invoer == 'j' THEN LET
```

```
    getal=getal+bijdrage
```

```
END DEFine vraag_kaart
```

```
:
```

```
PROCedure toon_kaart(kaartnummer)
```

```
    LOCAL loopteller
```

```
    OPEN #5,screen_def$
```

```
    PAPER #5,0 : INK #5,2
```

```
    AT #5,5,5
```

```
    SELECT ON kaartnummer
```

```
        = 1:
```



```

FOR loopteller = 0 TO 62 STEP 2
  PRINT !loopteller+1!
END FOR loopteller
= 2:
FOR loopteller = 0 TO 60 STEP 4
  PRINT !loopteller+2!loopteller+3!
END FOR loopteller
= 3:
:
:
:
END SELECT
close #5
END DEFine toon_kaart
:
DEFine FuNction invoer
  LOCAL toets$
  REPEAT toets
    LET toets$ = INKEY$(-1)
    IF toets$ INSTR 'jJnN' THEN EXIT toets
  END REPEAT toets
  RETURN toets$
END DEFine invoer

```

Hierbij wil ik het volgende kwijt. Ik geef ten eerste toe dat het minder overzichtelijk is wat er in toon\_kaart gebeurt voordat de getallen op het beeldscherm verschijnen. Dit komt doordat ik een algorithme heb gebruikt, in plaats van een string om de getallen te maken. Dit is alleen maar om aan te tonen dat een probleem vaak meerdere oplossingen geeft.

Verder heb ik van de PROCEDURE invoer een FuNction gemaakt. In feite is het enige dat deze PROCEDURE deed, een waarde afleveren. Deze werd in 'toets\$' terug gegeven aan het hoofdprogramma. Als er maar 1 waarde hoeft te worden teruggegeven, is een FuNction echter de aangewezen weg.

Het maken van een procedure 'toon\_kaart' kan een aanzienlijke hoeveelheid intikwerk besparen doordat er niet steeds 'at #5,5,5' etc. hoeft te worden ingetypt. Zou een bepaalde kaart meer dan 1 regel nodig hebben, dan is het nodig opnieuw het 'AT' commando te geven. Misschien is het dan eenvoudiger om een ander window op te geven. Dit window zou normaal gesproken overal door het programma opduiken, en dus overal aangepast moeten worden. Hier hebben we de definitie in een string gezet, die nmalig, bijvoorbeeld in een procedure 'init' een waarde gegeven wordt.

*Verder kan ik welnig kritiek hebben; overzichtelijk, procedures die met '::' geschieden zijn, overvloedig commentaar: ga zo door.*



## Errata

### alternatief karakter fount voor QUILL.

In de Quasar van februari 1987 zag ik dat er een regel tekst was weggefallen. Bovendien zag ik dat ik de verkeerde versie van zowel artikel als programma had ingestuurd. In het artikel moet staan:

Op adres 27604 tot en met 27617 staat het volgende stukje code:

27604	\$C407 AND.B D7,D2
27606	\$6706 BEQ.S LAB_1
27608	\$C311 AND.B D1,(A1)
27610	\$93C9 SUBAL A1,A1
27612	\$6012 BRAS LAB_2
27614	LAB_1 \$1411 MOVE.B (A1),D2
27616	\$C311 AND.B D1,(A1)

als dat veranderd wordt in :

27604	\$C311 AND.B D1,(A1)
27606	\$C407 AND.B D7,D2
27608	\$6706 BEQ.S LAB_1
27610	\$43FAxxxx LEA xxxx(PC),A1
27614	\$600E BRAS LAB_2
27616	LAB_1 \$1411 MOVE.B (A1),D2

dan hoeft op de plaats van xxxx allen nog maar de offset naar het zelf gedefinieerde normale karakter fount ingevuld te worden.

Het bijhorende SuperBasic programma moet als volgt gewijzigd worden:

```

470 DIM old_code%(6), new_code%(6), offset%(1)
540 DATA -15353, 26374, -15599, -27703, 24594,
      5137, -15599, -18149
545 DATA -15599, 15353, 26374, 17402, 24798,
      24590, 5137, 67349
585 patch = check_patch(27604, new_code%)
595 IF NOT check_patch(27604, old_code%) THEN
      escape 3
655 IF NOT patch THEN patch_quill
      27604, new_code%
680 IF patch THEN patch_quill 27604, old_code%

```

In de februari versie ging de nieuwe QUILL de mist in als je op een regel normal, lower en upper door elkaar gebruikte. Dit ging alleen op het scherm mis, bij het uitprinten van de tekst ging het wel goed. Met de hier gegeven verbeteringen gaat alles wel goed.

*Dat hoop ik tenminste.*

Hans Kraakman  
020-230795



Leliden, 2-4-1987

Beste heer den Breems,

Bij deze deel ik U mede dat ik mijn lidmaatschap opzeg per 1 april 1987 i.v.m. de aanschaf van een .....

Voorts wilde ik U en Uw collega's van Sin-QL-air danken voor de vele nuttige informatie die U via Quasar verstrekt heeft. Helaas ben ik niet produktief geweest met bijdragen aan Quasar, maar bijgaand treft U een kookboekprogramma voor Archive (1e versie; AVT-120 printer) aan. Misschien dat iemand hier iets aan heeft. Op de cartridge staat eveneens een Quill-file genaamd 'ADVERT' (van advertentie). Ewald Faber.

Te koop aangeboden:

Sinclair QL, Ancona monochroom monitor, 24 cartridges (20 lege), parallelle printer interface, alle Quasars tot april 1987, Sinclair QL gebruikersboek, "the Working Sinclair QL". Alles in 1 koop. Vraagprijs fl 450,00. Ewald Faber, Oude Vest 33CD, 2312 XR Leiden, tel. 071-134665.

#### HULP GEZOCHT BIJ ARCHIVE(REN)

Ik ben bezig aam een onderzoek t.b.v. een scriptie. Daarvoor moet ik interviews analyseren. Nu wil ik de genummerde categorieën zodanig in Archive opnemen, dat ze in verschillende combinaties kunnen worden opgehoest. Wie kan mij leren hoe dat moet?

Joy Bakker - 070 - 281710 (privé)  
070 - 406098 (kantoor)



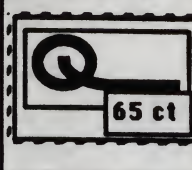
#### MEER OVER DE TANDY PLOTTER CGP-115.

Nadat ik in de vorige Quasar een stukje schreef over de Tandy plotter voor een leuke opruimingsprijs, ben ik bedolven onder een stortvloed van telefoontjes van mensen uit het hele land. De vragen die gesteld werden liepen nogal uiteen, maar de meeste mensen vroegen hoe de aansluiting moest geschieden. Ik zal dus nu het aansluitschematje hierbij afbeelden voor diegenen die het nog niet hebben uitgevonden. De instelling van de computer is als volgt;

1. OPEN #3, SER1HC
2. BAUD 600

ingezonden

mededelingen



#### ARCHIVE-VRAAG.

Bestaat er een routine of een POKE, zodat Archive gelijk in MODE 0 begint, dus zonder eerst zijn eigen standaard scherm te maken met de commando's e.d.?

#### ZWITSERS SYSTEEM?

Is er misschien iemand die weet of er een 'Zwitsers Systeem' is voor de QL?

Henk ten Bos De Laan 80g, 9981 GR

UITHUIZEN

Tel: 05953 - 1299

#### AANGEBODEN.

Ik heb voor de liefhebber 2 Atari SF 354 floppydrives te koop. (SSDD) Opslagcapaciteit is 360K per drive. De prijs is fl. 350,- p.s. incl. voeding en kabel naar QL diskinterface. Wel graag een vergoeding voor de portokosten.

Fred Brands - 053-772211 na 18.00 uur

Te koop aangeboden wegens aanschaf QL:

ATARI 520ST met diskdrive, beeldscherm, muis, TOS in ROM, 30 diskettes, tekstverwerking, spreadsheet, database, basic, fortran, C, etc. f1250,-

Telefoon: 08380 - 10318

TE KOOP: 256 K geheugenuitbreiding van Miracle Systems, direct aan te sluiten via de expansion port.

Fl 250,- Siebren Wierstra Tel: 02289-1284

Mensen die de printer niet aan de praat krijgen met bijv. Quill en Archive kunnen bij mij de juiste printerdriver krijgen. Stuur in dat geval een MDV of 3'5" floppy op. Het is zelfs mogelijk een 4-kleuren screendump te maken met deze plotter, je moet alleen geen haast hebben, e.e.a. duurt ruim 8 uur, maar het resultaat is er dan ook naar. Een lid uit Uitgeest schreef deze screendump routine en heeft mij toegezegd deze in de Quasar te publiceren. Nog even wachten dus! Het aansluitschema van de Tandy plotter is als volgt;

DIN SER1

1-----NIET GEBRUIKT

2-----3

3-----6

4-----4

F.Brands



# Software

BESPREKING

PROF. ASTROLOGER & ASTRONOMER

Astrologie programma t.b.v. het trekken van de geboorte-, dag- en jaarhoroscoop. Bovendien kan een compatibiliteitstest tussen twee personen worden uitgevoerd. Het astronomie programma geeft gegevens van de planeten op een bepaald tijdstip. Ik ben een leek wat astrologie betreft en hecht er geen waarde aan. Ik heb het programma gekocht omdat mijn vrouw en ik toch wel geïnteresseerd waren. Afgaande op besprekingen in andere bladen (o.a. QL World, sept '86) en de astroloog van DP is het pakket voor een astroloog zeer goed bruikbaar.

## Gegevens.

Uitgever: Digital Precision (afgekort DP) Prijs :  
69.95 ; DM 176.00 bij Uwe Fischer  
in Hamburg

Versie : 1.4

Omvang : 1007 sectoren, programma's en  
gegevens (29 files)

Medium : 3.25 inch diskette (of 5 cartridges)

Prog's : ASTROLOGER (SuperCharged, de rest  
evenzo)

DEFAULTS\_EDITOR,  
TEXT\_EDITOR en  
ASTRONOMER.

Manual : 150 pagina's rood papier.

## Gebruiksvriendelijkheid.

Doe de diskette in de drive en tik 'LRUN  
FLP1\_BOOT' in. Er verschijnt een menu met de 4  
programma's en de keuze 'SuperBASIC', na de  
keuze voor ASTROLOGER wordt het programma  
uitgevoerd.

De aanwijzingen op het scherm spreken voor zich,  
na het ingeven van de geboortedatum, tijdstip,  
lengte en breedte graad wordt gevraagd of men  
een radix, transit of progression wil. Daarna moet  
aangegeven worden welke gegevens gewenst  
zijn. Vervolgens worden de volgende gegevens  
getoond: Overzicht van de planeten en  
berekende gegevens, Grafische weergave  
(wagenwiel), Aspecten en de Interpretatie teksten  
(bij mij 277 regels radix).

De schermen waar men om een interpretatie kan  
vragen is bij de radix het 'save'-scherm en bij de  
andere het 'aspects'-scherm. Dit schept in het  
begin enige verwarring.

Nu is het raadzaam de manual te bekijken en de  
zaken stap voor stap na te lopen. De manual is  
goed samengesteld. De volgende onderwerpen  
komen o.a. aan de orde: Programma's en files,  
Inleiding in de astrologie, Gebruiksaanwijzingen  
voor de programma's, Kaartje met de tijdzones,  
Lijst van steden met lengte en breedte graden  
(o.a. The Hague en Heerlen).

De manual bevat enkele drukproblemen bij  
tabellen, zodat er goed gelezen en gepuzzeld  
moet worden. Een probleem is wel het engels van  
de manual, daar deze doorspekt is met  
(astrologisch) technische termen. Bovendien  
worden de planeten en aspecten op het scherm  
met symbolen weergegeven wat de behoefte aan  
een papiertje met verklaringen wekt, deze  
bevinden zich verspreid in de manual en helaas  
werkt bij de DEFAULTS\_EDITOR de  
<CTRL/SHIFT/F2> niet om een hardcopy van een  
scherm met alle symbolen te maken.

## Mogelijkheden.

Hier volgt een opsomming van de mogelijkheden  
om de geïnteresseerde lezer (astroloog) een  
indruk te geven van de bruikbaarheid van de  
programmatuur.

BD- en BDX-files: BD-files bevatten de geboorte-  
gegevens van 1 persoon, daar dit veel loze ruimte  
op schijf veroorzaakt, kunnen deze worden  
gekomprimeerd in een BDX-file.

De in een BDX-file voorkomende namen kunnen  
op het 1e scherm worden getoond d.m.v.  
<D>,<. >,<ENTER>. Een goed reactie vermogen is  
gewenst om het scherm te 'freeze' met  
<CTRL/F5>.

Tekstuitvoer: Scherm, printer, file en er is een  
AUTOPRINT faciliteit t.b.v. dag- en jaarhoroscopen  
van enkele dagen en jaren.

Grafische symbolen: Grafische symbolen  
worden tussen de tekst geprint bij de  
planetengegevens en de aspecten. Met de  
DEFAULTS\_EDITOR kan men dit fenomeen  
uitschakelen.

FILES-INDEX file: In de FILES-INDEX wordt  
aangegeven welke tekst-files gebruikt worden  
t.b.v. de interpretaties. Men kan de volgorde, soort  
horoscoop en drive opgeven.

Hulzen-systemen: 7 mogelijkheden, default is  
Placidus.

Printer settings: Instelbaar met de  
DEFAULTS\_EDITOR. Indien men de tekst  
naderhand met QUILL wil bewerken is het handig  
de defaults voor de 'SP1000' te gebruiken.

Symbolen: Kunnen met de  
DEFAULTS\_EDITOR worden gewijzigd.

Orbs: Toegestane afwijking van de aspecten.  
Instelbaar zijn, de graden m.b.t. de planeten, het  
percentage m.b.t. de 'cusps' van de huizen en  
percentage van de orb bij de geboorte t.b.v. dag-  
en jaarhoroscoop en compatibiliteitstest.

TEXT\_EDITOR: T.b.v. BD- en BDX-files, de



FILES\_INDEX file en de interpretatie-files. DP zegt, 'gebruik liever je eigen', zoals de Metacomco Editor of QUILL (Alleen tekstregels afgesloten met een <CR>, geen \_DOC file maar een \_LIS file).

**Interpretatie-files:** Bevatten de tekstfragmenten die geselecteerd worden a.h.v. de stand van de planeten. De tekstfragmenten worden voorafgegaan door konditieregels waaraan moet worden voldaan. Ik zal trachten hier de syntax weer te geven, deze is niet zo in de manual aangegeven.

```
<konditie>:=
  (*<konditie>)*P<planeetnr>:<konditie>
  [or(*<konditie>)*P<planeetnr>:<konditie> ]]
```

or = impliciete operator OR, niet intikken !

Bij \*P-regels mag in een <konditie>, <planeet> door '\$' worden vervangen. Bij testen wordt de '\$' vervangen door de met de planeetnummers overeenkomende planeten (OR-operatie).

```
<konditie>:=<SPACE><konditie>[AND<konditie>[.]]<CR>

<konditie>:= { <planeet> [NOT] in {<teken> | <huis>}
               |<planeet> [NOT] <hoek> <planeet>
               |<planeet> [NOT] in {RAD | OTH} <huis>
               | {RAD | OTH}<planeet> [NOT] <hoek>

<planeet>
               | <planeet> [NOT] <hoek> {RAD | OTH}

<planeet>}
```

'in' is voor de duidelijkheid aangegeven.

```
<planeet> := SUN | MOON | LUNar node | etcetera
<planeetnr> := combinatie van 1 - 10 cijfers van 0 - 9
<teken> := ARies | TAUrus | GEMini | etcetera
<huis> := FIRst | SEConD | THIRd | etcetera
<hoek> := CONJunction | OPPosition | SSE | etcetera
```

## Resultaten

De geboortehoroscoop bestaat uit de volgende zaken, overzicht planeetgegevens, aspecten, hardcopy van wagenwiel en interpretatietekst.

Bij dag- en jaarhoroscopen heeft men de aspecten t.o.v. planeten bij de geboorte. Bij de compatibiliteitstest heeft men het 'cross-test'-wagenwiel en de cross-aspecten. De gegevens zien er goed verzorgd uit. Wil men een goed verzorgde tekst, dan is dit te realiseren met QUILL. De teksten zijn in het engels en bevatten veel een opsomming van kreten, alle uitersten komen voor. Men zal zelf de interpretatie moeten aanvullen met andere zaken, zoals de invloed van de 'retrograde' planeten, het aantal planeten in een groep huizen en het aantal positieve en negatieve planeten. Ik stuur alles naar een file, behalve het wagenwiel, hier maak ik een hardcopy van.

Het printen van grafische tekens tussen de tekst heb ik uitgeschakeld, daar de zaak in de soep liep. Dit laatste moet ik nog uitzoeken voor mijn STAR-SG10. Het direkt printen van deze file geeft geen mooi resultaat. Het printen begint helemaal links

op het papier en er zijn geen form-feeds om de zoveel regels. Denk aan graden teken!

## Astronomie programma

Het astronomie programma komt overeen met het astrologie programma. De volgende zaken worden getoond: Tabel met planeetgegevens, De aangezichten van de binnenste planeten, Het planetarium met de planeten en Het zonnestelsel.

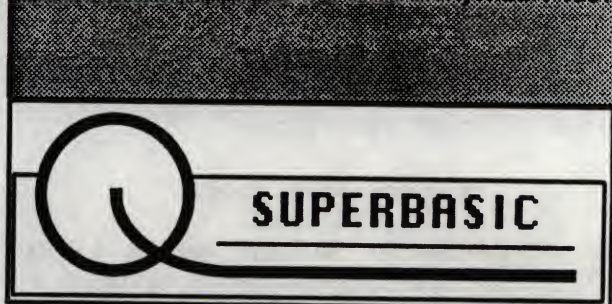
**Extra mogelijkheden zijn:** Weergave van hoeken in graden minuten en seconden of graden minuten en decimalen, 5 coördinaten systemen kunnen worden ingesteld.

Het programma heeft de volgende files nodig: ASTRO\_TABLE, DEFAULTS en HARDCOPY\_DRIVER. Een onvolkomenheid is dat bij een hardcopy van het scherm met de aangezichten van de planeten, deze toch als eieren worden weergegeven.

## Konklusie

De programmatuur ziet er professioneel uit, goed afgewerkt met nauwkeurige berekeningen. De interpretatiefiles met de kondities bieden zeer veel mogelijkheden, het riekt naar een EXPERT SYSTEEM !! Deze programmatuur is aan te raden voor de amateur astroloog, voor de professionele astroloog zie ik in dit produkt een onmisbaar hulpmiddel. Hij kan snel de berekeningen uitvoeren en heeft nu meer tijd voor het interpreteren. Het astronomie programma lijkt meer bedoeld voor iemand die met een kijker naar de planeten kijkt.

Ed Vermeulen, maart 1987



## 'Kleintje' Datum, PI en Dertien. door Sieben Wierstra.

Het nu volgende programma berekent op welke dag een bepaalde datum viel of valt:

```
100 PRINT "Voer in: Datum, maand en jaar..."
110 PRINT "Datum?... ";
120 INPUT D
130 PRINT "Maand?... ";
140 INPUT M
150 PRINT "Jaar?... ";
160 INPUT J
170 F=J+D+3*M-3
```



```

180 X=INT(SNG(M-3)/2)
190 J=J+X
200 F=F-(X+1)*INT(.4*M+2.3)
210 F=F+INT(J/4)-INT(3*INT(J/100)/4)-1
220 F=F-7*INT(F/7)
230 A$="ZATERDAG.ZONDAG...MAANDAG
    ..DINSDAG..WOENSDAG.DONDERDAGVRIJ
    DAG.."
240 PRINT:PRINT "Het viel op een "; A$(F*9 to
    (F*9)+9)

```

(UIT: KIJK Maart 1984, Ingezonden door Hans Stavleu uit Leiden)

Denk erom dat je in A\$ van regel 230 het juiste aantal puntjes (of spaties) neerzet.

Het volgende programma berekent de waarde van PI. Naarmate het programma langer draait, benadert deze waarde het getal PI steeds meer...

```

90 WINDOW 512,256,0,0:PAPER 0:INK 4:CLS
100 PRINT "PI - BEREKENEN"
111 AT 4,0:PRINT "PI is ongeveer ."
112 at 6,0:print "Rekentijd ."
113 at 8,0:PRINT "Aantal berekeningen:"
110 DATUM= DATE
120 T=1:P=0:I=0
130 P=P+1/T-1/(T+2)
140 T=T+4:I=I+1
150 AT 4,21:PRINT P*4
151 A=DATE-DATUM
152 AA=INT(A/3600):A=A-(AA*3600)
153 AAA=INT(A/60):A=A-(AAA*60)
154 A$=A:IF LEN(A$)<2:A$="0"&A$
155 AA$=AA:IF LEN(AA$)<2:AA$="0"&AA$
156 AAA$=AAA:IF LEN(AAA$)<2:
    AA$="0"&AAA$
160 AT 6,21: PRINT aa$;aaa$;a$
170 AT 8,21: PRINT I
180 GOTO 130

```

(UIT: KIJK-EXTRA Oktober 1985)

Als je bang bent voor Vrijdag de 13 (heb je de film soms gezien....) dan is hier het programma dat je kan vertellen wanneer je op moet passen:

```

100 cls
110 PRINT "VRIJDAG DE 13/"
120 INPUT "Welk jaar?... ";J
130 B=J/28
140 C=INT(B)
150 D=B-C
160 A=28*D
170 E=A+.5
180 A=INT(E)
190 PRINT:PRINT
200 IF A=24:PRINT "JANUARI, APRIL EN JULI"
210 IF A=25 OR A=3 OR A=8 OR A=14:PRINT
    "SEPTEMBER EN DECEMBER"
220 IF A=26 OR A=9 OR A=15 OR A=20:PRINT

```

```

"JUNI"
230 IF A=27 OR A=10 OR A=21:PRINT
    "FEBRUARI, MAART EN NOVEMBER"
240 IF A=0 OR A=6 OR A=17 OR A=23:PRINT
    "MEI"
250 IF A=1 OR A=7 OR A=18:PRINT "JANUARI
    EN OKTOBER"
260 IF A=2 OR A=13 OR A=19:PRINT "APRIL EN
    JULI"
270 IF A=4:PRINT "MAART EN NOVEMBER"
280 IF A=5 OR A=11 OR A=22:PRINT
    "AUGUSTUS"
290 IF A=12:PRINT "OKTOBER"
300 IF A=16:PRINT "FEBRUARI EN AUGUSTUS"
310 PRINT:PRINT
320 GOTO 120

```

(UIT: KIJK November 1985, Ingezonden door Martin Meljerink uit Laren (Gld)) Let even op regel 170:De opdracht INT rondt altijd naar beneden af. Dus INT(1.9) wordt 1. Wil je wiskundig afronden, dus.0 - .4 naar beneden.5 - .9 naar bovendan moet je bij het af te ronden getal gewoon 0.5 optellen. Dan kom je wel goed uit.

Dit waren een paar voorbeelden van programma's die geschreven zijn op een andere computer, maar die het met een beetje aanpassen op vrijwel elke computer doen. Deze programma's komen uit de tijd dat ik nog een ZX Spectrum had.

Siebrun Wierstra,Knokkel 88,1619 AK Andijk

Vooruitblik...



QL or not to QL.

Ik heb aangeboden om voor de Quasar een aantal artikelen te schrijven, of desgewenst te vertalen. Het is natuurlijk niet mogelijk om het iedereen naar de zin te maken, maar toch is het erg belangrijk dat ik weet wat er bij de leden voor belangstelling is. Iedereen heeft natuurlijk een boel voorkeur voor 1 specifiek onderwerp, en het is in principe niet onmogelijk om daar eens wat aandacht aan te besteden.Het volgende staat voor de volgende nummers op stapel;

1. Recensies van de programma's uit de cartridgeservice. (Uitvoerige beschrijving van de soms erg leuke programma's en utility's)
2. Zeer uitgebreid verhaal met voorbeelden etc. over Artificial Intelligence (AI). Kan een computer denken als een mens ? Soms vertoont een QL haast menselijke trekjes, zoals het geen zin meer hebben. Wat is wel- en wat is niet mogelijk? Lees het allemaal in de volgende QUASARS.

Ik zou graag van leden horen waar hun problemen of vragen liggen, ik probeer dan graag voor een passend artikel te zorgen.  
Fred Brands.053-772211





## TANDATA MODEM....



Wilt u een tandata modem kopen dan is het verstandig om hem bij DATA-SKIP (01820-20581) te kopen. Daar betaalt u maar 349,- en hebt u hem de volgende dag. Bij Komin betaalt u F599,- en moet u DRIE weken wachten wegens langzame levering uit Engeland ????

### HET AANSLUITEN VAN HET TANDATA MODEM.

Dit modem wordt geleverd met een engelse telefoonplug. Voor het aansluiten zijn twee systemen :de snelle en de nette. Voor beide systemen heeft u een telefoonplug, 2 kabelschoentjes (oog met rood kapje) en een 2-draads telefoontoestel (Ik heb zelf een HEMA telefoon van F15.--) of een PTT telefoon waar u de blauwe en de rode draad van gebruikt.

#### DE SNELLE METHODE

Knip de telefoonplug & ongeveer 20 cm kabel van het modem, strip allebei de einden af soldeer er kabelschoentjes aan (goed vastknijpen en vast solderen), maak de rode draad van de modemkabel aan A van de telefoonplug vast en de witte aan de B. Soldeer uw telefoontoestel aan de rode en de witte draad van de engelse telefoonplug vast en stop dit in uw modem in de aansluiting gemerkt met PHONE (aan de rechterzijde van het modem)

#### DE NETTE METHODE

Neem een kleine schroevendraaier, een priem oid. (met sommige scharen gaat het ook) en steek die tussen een van de koperen contactjes van de modemplug (aan de voorkant, wip hem omhoog en haal hem er dan uit. Doe dit met alle 4 de contactjes (bewaar ze wel). Geef als dit gebeurd is een ruk aan de plug en hij is er af (deze methode werkt ook bij SER & CTL pluggen). Neem uw 2 draads-telefoontoestel haal evt. de plug en evt. kabelschoentjes eraf en verwijder de draadisolatie over een lengte van ong. 1 cm (de lengte komt niet zo nauw). Vertin deze draadeindjes en neem dan de losgemaakte plug. Steek een van de einden in de plug en mik zodanig dat het eindje in de meest linkse of meest rechtse aansluiting terecht komt. Ziet u het draadeindje in het tweede (van links of rechts) gleufje? Steek dan een van de contactjes in die gleuf en druk hem stevig aan. Doe dit ook met de andere maar dan in het andere (tweede) gaatje. Mocht het niet lukken vertin de draadjes dan opnieuw, het ligt waarschijnlijk aan de dikte van de draadjes. Strip de modemkabel een eindje af en SOLDEER KABELSCHOENTJES aan de rode en de witte draad. Eerst stevig aanknijpen

en dan vast solderen. Neem dan een telefoonplug en maak de rode draad aan A vast en de witte aan B. Steek uw telefoon met aangepaste plug in de PHONE aansluiting aan uw modem. (aan de rechter zijde)

### HOE WERKT U MET HET MODEM.

Laad het bijgeleverde programma QCONNECT en toets 3 x 1 (Viditel inloggen). Toets een nummer van een viditel-achtig bulletin board & ENTER (Viditel 06-8422 - Hobbytel 035-231513 beide 24 uur per dag) en druk op F2. Als het goed is krijgt u het openingsbeeld en moet u een toegangscode intoetsen (Hobbytel gastcode 1111123456789 zonder enter) U kunt zo alle 1200/75 baud BBS'en bellen behalve viditel BBS'en die op een CBM 64 werken (mbase bbs programma), bij deze moet u in het menu wat u krijgt na het intoetsen van het telefoonnr. de parity OFF zetten en het aantal bits op 8 (normaal 7 bits met parity) 2 tekens die u bij viditel-achtige systemen nodig heeft zijn \* en #. Niet viditel BBS'en kunt u in VT100 of viditel gebruiken (VT100 heeft meer mogelijkheden). Zorg dat er geen telefoontoestellen parallel staan aan uw modem (trek de plug van evt slaap- en huiskamer toestellen eruit), en gebruik dus enkel de telefoon die in uw modem zit als u aan het bellen bent met uw modem (deze telefoon wordt uitgeschakeld als het modem wat u belt on-line gaat).

### HCC-FIDO

Het bellen met fido is een verhaal apart. Het makkelijkste is om uw modem in te stellen op 1200/75 parity off 8 bits & flow control op XON. bel dan fido (met de hand) en als fido zijn carrier geeft druk dan op space & enter. Nu heeft fido 3 verschillende carriers 300 - 1200 & 2400 baud en u moet dus ontdekken welke de 1200 carrier is. Dit is een kwestie van proberen. Over enige tijd wordt de volgorde bij elke fido gelijk maar nu word er nog mee geëxperimenteerd. Als alles klopt gaat uw modem nu on-line. Ziet u allerlei onzin dan heeft u de verkeerde. Druk dan op F2 en dan op 2 (hang-up). Draai het nummer dan opnieuw en probeer een andere carrier. Wordt uw scherm zwart (modem on-line) en ziet u niets dan is het goed, druk dan een aantal keren op enter en het openingsbeeld verschijnt.

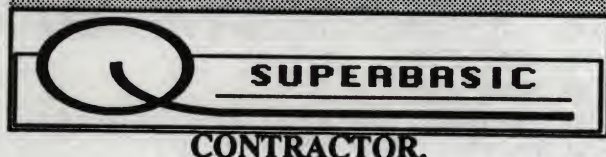
### QL INFO IN BBS'en

Werkt alles naar behoren bel dan eens naar RvV-data en ga dan naar pagina \*140# (Info Leverancier FT-TEL). Hierin staat QL info (als ik het toegeleverd krijg tenminste, bellen of schrijven dus). Ook kunt u naar DINDAT bellen. Ook daarin is QL info te vinden. RvV data kunt u bellen op telfnr. 01804-27897, 24 uur per dag in viditel-mode. DINDAT kunt u bellen op telfnr. 01672-2076 op werkdagen van 7.30 tot 17.00 en op



maandagavond van 19.00 tot 22.00 in viditel of VT 100 met 8 bits zonder parity. U moet wel lid worden maar dat is gratis en er wordt duidelijk aangegeven hoe, als u als gast inlogt (gastnummers in openingsbeeld) Mocht u info hebben voor deze twee bestanden of moeilijkheden met het aansluiten van of werken met het TANDATA modem dan kunt u me bellen of schrijven :

Frank Troost, Steenbergseweg 74671 BC Dinteloord. 01672-2584 (spraak)  
Of laat een bericht achter in DINDAT (nr 4) of RvV-DATA (USER 5844).



```
1000 REMark 'CONTRACTOR' : PA Verhoef
      : versie 1.1
1010 REMark zorgt ervoor,dat programma
      verkort wordt
1020 REMark door alle overbodige spaties te
      verwijderen.
1030 REMark Bovendien worden woorden als
      DEFINE, RETURN enz. afgekort.
1040 REMark Teksten tussen 'quotes' en in
1050 REMark 'REMark' statements blijven
      onveranderd
1060 :
1070 CLEAR
1080 symbol$="!@#$%^&*()-+=~\:'<>./,!"&""
1090 alfabet$="abcdefghijklmnopqrstuvwxyz": REMark
      digit$="0123456789"
1100 PAPER 2:CLS:CLS#0:CSIZE 3,1:AT 4,8:
      PRINT" > CONTRACTOR <":CSIZE 0,0
1110 INPUT#0,"Source_file? ";object$
1120 INPUT#0,"Dest'_file? ";source$
1130 OPEN_IN#4,object$
1140 OPEN_NEW#5,source$
1150 REPEAT lees_regel
1160 IF EOF(#4)THEN EXIT lees_regel
1170 INPUT#4,regel$
1180 nquote regel$
1185 PRINT#5
1190 END REPEAT lees_regel
1200 CLOSE#4:CLOSE#5
1210 STOP
1220 :
1230 DEFINE FuNction instr(n,hoofd$,item$)
1240 LOCAL p
1250 p=item$ INSTR hoofd$(n TO)
1260 IF p=0 THEN RETURN 0
1270 RETURN p+n-1
1280 END DEFINE
1290 :
1300 DEFINE PROCEDURE nquote(a$)
1310 REMark zoek niet tussen aanhalingstekens
1320 LOCAL nquote,nquote2,n,b,l,rem_$
1330 rem_$="":regelnr%=a$
1340 b=" "INSTR a$
```

```
1350 REPEAT 1
1360   nquote=instr(b,a$, ""):IF nquote=0 THEN
      nquote=LEN(a$)+1
1370   nquote2=instr(b,a$, ""):IF nquote2=0 THEN
      nquote2=LEN(a$)+1
1380   IF nquote2<nquote THEN
      nquote=nquote2:REMark kleinste van 2 nemen
1390   IF nquote>LEN(a$) THEN
      nquote2=0
1400   nquote2=0
1410   ELSE
      nquote2=instr(nquote+1,a$,a$(nquote))
1420   IF nquote2=0 THEN nquote=LEN(a$)+1
1430   END IF
1440   b$a$(b TO nquote-1)
1450   zoekstatement b$,rem_$
1460   IF regelnr%>0 THEN PRINT#5,
      regelnr%:regelnr%=0
1470   PRINT#5, b$;
1480   IF nquote=LEN(a$)OR nquote2=0 OR
      rem_$<>" THEN EXIT 1
1490   PRINT#5,a$(nquote TO nquote2);
1500   b=nquote2+1
1510   IF b>LEN(a$) THEN EXIT 1
1520   END REPEAT 1
1530 IF rem_$=" "THEN RETURN
1540 PRINT#5,"REM":rem_$ (7 TO);
1550 IF nquote<LEN(a$)THEN PRINT#5, a$(nquote
      TO);
1560 END DEFINE nquote
1570 :
1580 :
1590 DEFINE PROCEDURE zoekstatement
      (hoofd$,rem_$)
1600 LOCAL n,hoofdlius,k,b,end_fl$
1610 IF alleen_dp(hoofd$) THEN hoofd$=" ":RETURN
1620 split_rem hoofd$,rem_$
1630 verander_then hoofd$
1640 step_punt hoofd$
1650 b=1
1660 REPEAT hoofdlius
1670   FOR n=b TO LEN(hoofd$):IF hoofd$(n)
      INSTR alfabet$&&"=" THEN EXIT n
1680   end_fl$="l"
1690   sla_end_over hoofd$,n,end_fl$
1700   REMark PRINT#0,hoofd$,LEN(hoofd$),n
1710   REMark volgende 2 regels versnelt de
      uitvoering(hoop ik...)
1720   IF hoofd$(n)INSTR"RDIRSL" AND n<LEN
      (hoofd$)THEN
1730     IF hoofd$(n+1)INSTR"EAFO"THEN
1740       REMark let op de punten in de volgende
      regels
1750       IF instr(n,hoofd$,"RETURN ")=n THEN
      hoofd$ (n TO n+5)="RET."
1760       IF instr(n,hoofd$,"DATA ")=n THEN
      hoofd$(n+4)="."
1770       IF instr(n,hoofd$,"RESTORE ")=n THEN
      hoofd$(n+7)="."
1780       IF instr(n,hoofd$,"IF ")=n THEN
      hoofd$(n+2)="."
1790       IF instr(n,hoofd$,"REPEAT ")=n THEN
      hoofd$(n TO
      n+5)="REP."
```



```

1800 IF instr(n,hoofd$,"DEFine ")=n THEN
hoofd$(n TO n+6)="DEF"&end_fl$
1810 IF instr(n,hoofd$,"SELEct ")=n THEN
hoofd$(n TO n+6)="SEL"
1820 IF instr(n,hoofd$,"LOCal ")=n THEN
hoofd$(n TO n+5)="LOC"
1830 REMark enz. (eentje vergeten misschien.)
1840 END IF
1850 END IF
1860 k=instr(n,hoofd$,"GO ")
1870 IF k>1 THEN
1880 IF hoofd$(k-1)INSTR ")": " THEN
hoofd$(k+2)="I"
1890 END IF
1900 k=" PROCedure "INSTR hoofd$:IF k THEN
hoofd$(k TO k+9)=" PROC."
1910 k=" FuNction "INSTR hoofd$:IF k THEN
hoofd$(k TO k+8)=" FN."
1920 b=instr(n,hoofd$,".")
1930 IF b=0 OR b=LEN(hoofd$)THEN EXIT
hoofd$lus
1940 END REPEAT hoofd$lus
1950 verkort hoofd$
1960 IF rem_$<>" "AND hoofd$<>" " THEN
hoofd$=hoofd$&" "
1965 IF hoofd$="::"THEN hoofd$="::":REMark
opheffen van bug
1970 END DEFine zoekstatement
1980 :
1990 DEFine PROCedure verkort(hoofd$)
2000 LOCAL hulp$,af,n,naf,symbolool_fl
2010 hulp$=hoofd$:af=0:naf=1:symbolool_fl=1
2020 FOR n=2 TO LEN(hoofd$)
2030 IF hoofd$(n)INSTR symbolool$&" "THEN
2040 IF hoofd$(n-1)=" "THEN af=af+1
2050 IF hoofd$(n-1)INSTR symbolool$ THEN
symbolool_fl=1
2060 ELSE
2070 IF symbolool_fl AND hoofd$(n-1)=" "THEN
af=af+1
2080 symbolool_fl=0
2090 END IF
2100 naf=n-af:hulp$(naf)=hoofd$(n)
2110 IF hoofd$(n)="I"THEN af=af+1
2120 END FOR n
2130 hoofd$=hulp$(1 TO naf)
2140 FOR n=LEN(hoofd$)TO 1 STEP -1:IF NOT
hoofd$(n)INSTR " "THEN EXIT n
2150 hoofd$=hoofd$(1 TO n):IF hoofd$=" "THEN
hoofd$=""
2160 FOR n=1 TO LEN(hoofd$):IF
hoofd$(n)=" "THEN hoofd$(n)=" "
2170 END DEFine verkort
2180 :
2190 DEFine PROCedure verander_then(hoofd$)
2200 LOCAL n,l,symbolen$
2210 n=1:symbolen$=symbolool$&" "
2220 REPEAT l
2230 n=instr(n,hoofd$,"THEN ")
2240 IF n=0 THEN RETURN
2250 IF n>1 THEN
2260 IF hoofd$(n-1)INSTR symbolen$THEN

```

```

hoofd$(n TO n+3)=":"
2270 ELSE
2280 hoofd$(n TO n+3)=":"
2290 END IF
2300 n=n+1
2310 END REPEAT l
2320 END DEFine verander_then
2330 :
2340 DEFine PROCedure sla_end_over
(hoofd$,n,end_fl$)
2350 LOCAL k
2360 IF "END " INSTR hoofd$=n THEN
2370 hoofd$(n+3)="I":end_fl$=""
2380 FOR k=n+3 TO LEN(hoofd$)
2390 IF hoofd$(k)INSTR alfabet$ THEN EXIT k
2400 END FOR k
2410 ELSE
2420 RETURN
2430 END IF
2440 n=k:RETURN
2450 END DEFine sla_end_over
2460 :
2470 DEFine PROCedure step_punt(a$)
2480 LOCAL l,k
2490 k=1
2500 REPEAT l
2510 k=instr(k,a$,"STEP ")
2520 IF k=0 THEN EXIT l
2530 IF k>1 THEN
2540 IF a$(k-1)INSTR symbolool$&" "THEN
a$(k+4)="."
2550 ELSE
2560 a$(k+4)="."
2570 END IF
2580 k=k+1
2590 END REPEAT l
2600 END DEFine step_punt
2610 :
2620 DEFine PROCedure split_rem(hoofd$,rem_$)
2630 LOCAL l,n
2640 n=1
2650 REPEAT l
2660 n=instr(n,hoofd$,"REMark ")
2670 IF n=0 THEN RETURN
2680 IF n>1 THEN
2690 IF hoofd$(n-1)INSTR " ": rem_$=hoofd$ (n
TO) : hoofd$=hoofd$(1 TO n-1):EXIT l
2700 END IF
2710 n=n+1
2720 END REPEAT l
2730 END DEFine split_rem
2740 :
2750 DEFine FuNction alleen_dp(hoofd$)
2760 LOCAL n
2770 FOR n=1 TO LEN(hoofd$)
2780 IF NOT(hoofd$(n)INSTR " ")THEN RETURN 0
2790 END FOR n
2800 RETURN 1
2810 END DEFine alleen_dp

```



## All Play All Competitie



Bij het organiseren van een competitie waarbij elke deelnemer / elk team éénmaal tegen de anderen moet spelen, is het vaak een vervelend werk om uit te zoeken wanneer wie tegen wie moet spelen. Onderstaand programma'tje lost dit eenvoudig op. Allereerst moet het aantal deelnemers worden ingevoerd en vervolgens de naam van elke deelnemer. Daarna wordt de speelvolgorde voor iedereen berekend en moeten de aanvangstijd, speelduur per wedstrijd en de pauze tussen twee wedstrijden worden ingevoerd. De output kan worden geprint op scherm en printer (SER1). Wel is er de moeilijkheid dat voor veel deelnemers (meer dan 7) de scoretabel niet op het beeldscherm past. Maar daar hebben we dan ook een printer voor. Eerst wordt de scoretabel getoond en vervolgens de indeling per ronde met bijbehorende aanvangstijd.

<voorbeeld>

### SCORETABEL:

	1	2	3	4
1 J. Jansen	XX			
2 J. Bakker		XX		
3 S. Slager			XX	
4 E. Faber				XX

### INDELING PER RONDE

Ronde : 1			
19:30	J. Bakker - J. Jansen	...	
	E. Faber - S. Slager	...	
Ronde : 2			
19:55	J. Jansen - S. Slager	...	
	J. Bakker - E. Faber	...	
Ronde : 3			
20:20	E. Faber - J. Jansen	...	
	S. Slager - J. Bakker	...	

Henk ten Bos, De Laan 80g  
9981 GR UITHUIZEN  
Tel: 05953 - 1299

```

100 REMark
101 REMark | ALLPLAYALL COMPETITIE
102 REMark |
103 REMark | Henk ten Bos 03-87
104 REMark | Tel: 05953 - 1299
105 REMark
106 :
107 initialiseer
108 bereken
109 vraag_gegevens
110 CLS:score_tabel
111 CLS:indeling

```

```

112 CLS:STOP
113 :
114 DEFine PROCedure initialiseer
115 WINDOW#0,512,256,0,0:PAPER#0,0
116 CLS#0:WINDOW#0,460,40,26,20
117 WINDOW 460,160,26,70
118 PAPER#0,7:INK#0,0:BORDER#0,1,4
119 PAPER 0:INK 4:BORDER 1,7
120 CSIZE#0,2,1:CLS#0:CURSOR#0,100,10
121 PRINT#0,"All play all systeem";
122 CSIZE#0,0,0
123 CLS:AT 1,2
124 PRINT "Hoeveel deelnemers?"!;
125 INPUT a_sp:f1=a_sp+1:f2=a_sp^2+2
126 DIM nm$(f1,20),a(f1,f1)
127 DIM pa(f2),pb(f2)
128 CLS:AT 1,2
129 PRINT"Voer nu de deelnemers in:"
130 FOR i=1 TO a_sp
131 AT 4,5:PRINT FILL$(" ",20)
132 AT 4,2:PRINT "Nr ";i!;
133 INPUT nm$(i):l=LEN(nm$(i))
134 IF l<20 THEN
135 nm$(i)=nm$(i)&FILL$(" ",20-l)
136 END IF
137 END FOR i
138 v$="VRU"&FILL$(" ",16):casp=a_sp
139 IF a_sp MOD 2=1 THEN
140 a_sp=a_sp+1:nm$(a_sp)=v$
141 END IF
142 END DEFine initialiseer
143 :
144 DEFine PROCedure bereken
145 REMark -----
146 CLS:AT 1,2:PRINT"Een momentje..."
147 AT 2,2:PRINT"Indeling bepalen"
148 FOR i=1 TO a_sp:a(i,i)=0
149 FOR i=2 TO a_sp
150 a(1,i)=i-1:a(i,1)=i-1
151 END FOR i
152 l=2
153 FOR i=2 TO a_sp-1
154 k=a(i,1)+1
155 FOR j=2 TO a_sp-1
156 IF j<>i THEN
157 IF k=l THEN verh k,1
158 a(i,j)=k:verh k,1
159 END IF
160 END FOR j
161 a(i,a_sp)=l
162 verh l,2
163 END FOR i
164 FOR i=1 TO a_sp-1
165 a(a_sp,i)=a(i,a_sp)
166 END FOR i
167 a(a_sp,a_sp)=0
168 REMark -- Volgorde bepalen --
169 AT 3,2:PRINT"Volgorde bepalen"
170 n=1
171 FOR r=1 TO a_sp-1
172 i=1
173 REPEAT loop
174 IF i<a_sp THEN

```



```

175   FOR j=i+1 TO a_sp
176     IF a(i,j)=r THEN
177       pa(n)=i:pb(n)=j
178       n=n+1
179     END IF
180   END FOR j
181   i=i+1
182 ELSE
183   EXIT loop
184 END IF
185 END REPEAT loop
186 END FOR r
187 END DEFine bereken
188 :
189 DEFine PROCedure verh(x,p)
190 x=x+p
191 IF x>=a_sp THEN x=1
192 END DEFine verh
193 :
194 DEFine PROCedure vraag_gegevens
195 CLS:AT 1,2
196 PRINT"Aanvangstijd (uur)   :!";
197 INPUT au:AT 2,2
198 PRINT"Aanvangstijd (min)  :!";
199 INPUT am:AT 3,2
200 PRINT"Tijd per wdstr. (min) :!";
201 INPUT smin:AT 4,2
202 PRINT"Pauze na 'n ronde (min):!";
203 INPUT pmin:AT 6,2
204 PRINT"Display of SER1 (d/s) :!";
205 INPUT q$
206 IF q$="s" THEN OPEN#4,SER1:out=4
207 IF q$<>"s" THEN out=1
208 END DEFine gegevens_invoeren
209 :
210 DEFine PROCedure indeling
211 n$=am:r$=au
212 FOR r=1 TO a_sp-1
213   AT 2,2:PRINT#out,"Ronde :!r\
214   FOR i=1 TO a_sp DIV 2
215     IF i<>1 THEN
216       PRINT#out,"      ";
217     ELSE
218       tijd
219     END IF
220     ind=(r-1)*(a_sp DIV 2)+i
221     x=pa(ind):y=pb(ind)
222     bepaal_pr x,y
223   END FOR i
224   IF out=1 THEN wait
225 END FOR r
226 END DEFine druk_af
227 :
228 DEFine PROCedure bepaal_pr(x,y)
229 LOCAL a,b,s
230 a=x MOD 2:b=y MOD 2
231 SElect ON a
232 =0:SElect ON b
233   =0:IF x<y THEN s=0:ELSE s=1
234   =1:IF x>y THEN s=0:ELSE s=1
235 END SElect
236 =1:SElect ON b
237   =0:IF x>y THEN s=0:ELSE s=1

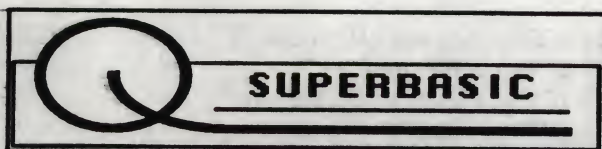
```

```

238   =1:IF x<y THEN s=0:ELSE s=1
239 END SElect
240 END SElect
241 IF s=0 THEN
242   PRINT#out,nm$(x)!-"!nm$(y)!!;
243 ELSE
244   PRINT#out,nm$(y)!-"!nm$(x)!!;
245 END IF
246 PRINT#out,"... "
247 END DEFine bep_en_print
248 :
249 DEFine PROCedure tijd
250 IF LEN(r$)=1 THEN PRINT#out,"0";
251 PRINT#out,r$;";";
252 IF LEN(n$)=1 THEN PRINT#out,"0";
253 PRINT#out,n$;";";
254 n$=n$+smin+pmin
255 REPEAT l
256   ng=0
257   IF n$>=60 THEN
258     r$=r$+1:n$=n$-60:ng=1
259   END IF
260   IF ng=0 THEN EXIT l
261 END REPEAT l
262 IF r$>=24 THEN r$=r$-24
263 END DEFine tijd
264 :
265 DEFine PROCedure score_tabel
266 v$=FILL$("- ",23+(easp+1)*3)
267 w$="  l":s$=FILL$(" ",26)
268 u$="XXI":CLS
269 PRINT#out,s$;
270 FOR i=1 TO easp
271   IF i<10 THEN PRINT#out," ";
272   PRINT#out,i!;
273 END FOR i
274 PRINT#out,\v$
275 FOR i=1 TO easp
276   PRINT#out,"l";
277   IF i<10 THEN PRINT#out," ";
278   PRINT#out,i!nm$(i)! "l";
279   FOR j=1 TO easp
280     IF j=i THEN
281       PRINT#out,u$;
282     ELSE
283       PRINT#out,w$;
284     END IF
285   END FOR j
286   PRINT#out,\v$
287 END FOR i
288 IF out=1 THEN wait
289 END DEFine scoretabel
290 :
291 DEFine PROCedure wait
292 AT 14,2:PRINT"<center>";
293 INPUT q$
294 END DEFine wait

```





FINDNAME\_bas, versie 1.1

### PA Verhoef

Dit programma zoekt naar een bepaald woord in prog/file. Het woord moet links door een karakter uit 'sym\$' begrensd worden, behalve als 'item\$' aan het begin vd regel gevonden wordt. Het woord moet rechts door een karakter uit 'sym\$' begrensd worden, behalve als 'item\$' aan het eind van de regel gevonden wordt. Indien het woord gevonden wordt en aan bovenstaande voorwaarden wordt voldaan, dan wordt het regelnummer waar het woord gevonden werd afgedrukt.

```

1110 CLEAR:REMark zonodig volgende regel
      veranderen /vervallen laten
1120 CLS#0:PAPER 2:CLS:CSIZE 3,1:AT
      4,10:PRINT"FINDNAME":CSIZE 0,0
1130 sym$=" !@#%&^*0_+~`N{ }[]; < > ? / , ' " & ""
1140 REMark cyfer$="1234567890"
1150 INPUT#0,"Source_file? ";object$
1160 INPUT#0,"Dest_file? ";source$
1170 INPUT#0,"Search item? ";item$
1180 OPEN_IN#4,object$
1190 OPEN_NEW#5,source$
1200 PRINT#5,"In het programma ";object$;" komt"
1210 PRINT#5,"""&item$&""!" voor als woord in de
      regels:"
1220 REPEAT lees_regel
1230 IF EOF(#4) THEN EXIT lees_regel
1240 INPUT#4,regel$
1250 regelnr%=regel$:REMark rest vervalt automatisch!
1260 vindnaam regel$,item$
1270 END REPEAT lees_regel
1280 CLOSE#4:PRINT#5:CLOSE#5
1290 STOP
1300 :
1310 DEFine PROCEDURE vindnaam(hoofd$,item$)
1320 LOCAL n
1330 n=1
1340 REPEAT loop
1350 IF n>LEN(hoofd$) THEN EXIT loop
1360 n=instr(n,hoofd$,item$)
1370 IF n>0 THEN
1380 IF fit(hoofd$,n,n+LEN(item$)-1,sym$,sym$)
      THEN
1385 PRINT#5,regelnr%,
1387 END IF
1390 ELSE
1400 EXIT loop
1410 END IF
1420 n=n+1
1430 END REPEAT loop
1440 END DEFine vindnaam
1450 :
1460 DEFine FuNction instr(n,hoofd$,item$)

```

```

1470 LOCAL p
1480 p=item$ INSTR hoofd$(n TO)
1490 IF p=0 THEN RETURN 0
1500 RETURN p+n-1
1510 END DEFine
1520 :
1530 DEFine FuNction fit(hoofd$,b,e,l$,r$)
1540 LOCAL ok_l,ok_r
1550 IF b<=1 THEN ok_l=1:ELSE ok_l=hoofd$(b)
      INSTR l$
1600 IF e>=LEN(hoofd$) THEN ok_r=1:ELSE
      ok_r=hoofd$(e) INSTR r$
1610 IF ok_l=0 THEN ok_l=hoofd$(b-1)INSTR l$
1620 IF ok_r=0 THEN ok_r=hoofd$(e+1)INSTR r$
1650 IF ok_r*ok_l THEN RETURN 1:ELSE RETURN 0
1660 END DEFine fit

```



Microsoft-Basic op de QL

Er is al veel software geschreven voor de CP/M computer en ook voor de IBM-PC's of compatibles. Verbazingwekkend is het hoeveel programma's daarvan in Basic geschreven zijn. Vooral in de USA zijn er veel Basic spelletjes voor de IBM-PC. Ook een van de beste Public-Domain Programma's, PC-Talk is in basic geschreven. Een goede assembler - CHASM- is in basic. Een kleine aanpassing is de moeite waard omdat deze programma's meestal gratis te verkrijgen zijn. De grafische en sound-commando's moeten met de hand individueel aangepast worden. En de volgende regels moeten eerst gemaakt worden:

- 1) Alle TAB's door TO vervangen.
- 2) Alle aanhalingstekens die commentaar voorstellen in REMark veranderen.
- 3) Indien het er niet in staat dan zal na THEN een GOTO aangebracht moeten worden.
- 4) Na alle NEXT commando's moeten de namen van de loopvariabelen opgenomen worden.
- 5) Voor alle ELSE commando's een dubbele punt zetten.

Dan zijn er nog drie constructie's die alleen met de hand veranderd moeten worden.

- 1) De constructie
 

```

10 WHILE voorwaarde
20
30 WEND
      moet in
10 REPEAT naam
20 If voorwaarde THEN EXIT naam
30
40 end REPEAT naam
      omgezet worden.

```



2) Wanneer bij een FOR-NEXT loop direct na het FOR statement het eerste commando van de lus volgt, dan moet het commando in een nieuwe regel gezet worden. In het andere geval geeft het problemen met de korte vorm van de FOR-NEXT lus (zie QL handboek)

Voorbeeld:

```
10 FOR I=1 TO 10: a=b**45
20
30 NEXT I
```

moet in

```
10 FOR I=1 TO 1011: a=b**45
20
30 NEXT I
```

veranderd worden.

3) Alle IF met teken zoals bijv:

```
IF a$ >= "a" THEN
```

moeten in ASCII code omgezet worden. Het bovenstaande voorbeeld wordt:

```
IF CODE (a$) >=97 THEN.
```

Dat ligt aan de de sorteervolgorde van de QL. Daar wordt niet de ASCII code vergeleken, maar de letters van het alfabet AaBbCc.. enz. (Handboek blz. 47) In de listing zijn de belangrijkste commando's omgezet, die men voor eenvoudige structuren nodig heeft. De functie INSTR (a\$,b\$) houdt geen rekening met grote of kleine letters. Bovendien zullen enige functie's niet het gehele scala aan mogelijkheden bedekken. Er zijn meestal alleen de belangrijkste parameters aangegeven. Dat is dan ook die vorm die het meeste voorkomt. Het is zeker mogelijk dat er nog verbeteringen aan te brengen zijn. En dat er nog wel één of ander commando ontbreekt!

Oosterbeek 13 febr. 1987. Dirk de Vogel.

```
5 REMark *****
10 REMark Aanpassing van Super Basic
20 REMark aan Microsoft Basic
30 REMark 1987 R.W.Gerling en
40 REMark Dirk de Vogel Versie 1.2
50 REMark *****
60 DEFine FuNction LEFT$(a$,b)
70 IF b=0 THEN RETurn ""
80 IF b>LEN (a$) THEN RETurn a$
90 RETurn a$(1 TO b)
100 END DEFine LEFT$
110 :
120 DEFine FuNction RIGHT$(a$,b)
130 IF b=0 THEN RETurn ""
140 IF b>LEN (a$) THEN RETurn a$
150 RETurn a$(LEN (a$)-b+1 TO)
160 END DEFine RIGHT$
170 :
180 DEFine FuNction INSTR(a$,b$)
190 RETurn b$ INSTR a$
200 END DEFine INSTR
210 :
```

```
220 DEFine FuNction MID$(a$,A,b)
230 IF b=0 THEN RETurn ""
240 IF A>LEN(a$) THEN RETurn ""
250 RETurn a$(A TO A+b-1)
260 END DEFine MID$
270 :
280 DEFine PROCedure COLOR (A,b)
290 INK A : PAPER b
300 END DEFine COLOR
310 :
320 DEFine PROCedure PLAY (a$)
330 REMark Is slechts een primitieve oplossing!
340 BEEP 20000,10,100,1000,10,5
350 END DEFine PLAY
360 :
370 DEFine FuNction VAL(X$)
380 RETurn X$
390 END DEFine VAL
400 :
410 DEFine FuNction STR$(X)
420 RETurn X
430 END DEFine STR$
440 :
450 DEFine FuNction SGN(X)
460 IF X>0 THEN RETurn 1 : ELSE IF X=0 THEN
    RETurn 0 : ELSE RETurn -1
470 END DEFine SGN
480 :
490 DEFine FuNction asc(X)
500 RETurn CODE (X)
510 END DEFine asc
520 :
530 DEFine PROCedure file$ (a$)
540 DIR a$
550 END DEFine file$
560 :
570 DEFine FuNction space$ (X)
580 RETurn FILL$(" ",X)
590 END DEFine space$
600 :
610 DEFine FuNction string$ (X,a$)
620 RETurn FILL$(a$(1),X)
630 END DEFine string$
```



**ARCHIVE UTILITY: 'ARCHIVIST'**

Uitgever: ARK Distribution, £29,95

Deze vierde, commerciële, applicatie van de Psion Archive 'Runtime module' (een kleinere versie van Archive die specifieke applicaties draait maar dan veel sneller - RWR) is net op de markt gebracht door ARK Distribution. De utility overbruggt een kloof tussen de gebruiker en de flexibele maar



moeilijke Archive 'query' taal. Archivist biedt de database gebruiker een aantal buitengewone hulpmiddelen om nieuwe applicaties te ontwikkelen. Tevens is het nu mogelijk zonder trucjes een bestaand bestand te gebruiken in een nieuwe applicatie. Ieder van 22 bestaande velden per record op een beeldscherm met 80 karakters per regel en 25 regels kan veranderd worden.

Archivist kan toegepast worden op bestanden als postadressen, huishoudelijke bestanden, postzegelverzamelingen, bibliotheekbestanden of een fotoregister, om er maar een paar te noemen. De commandos die het programma sturen worden met een druk op een toets uitgevoerd. Een nieuwe procedure maakt het mogelijk programma of data gegevens van microdrive of floppy te halen. Het Archivist beeldscherm maakt niet alleen records zichtbaar maar geeft ook informatie in een status regel zoals de bestandsnaam, aantal geheugenbytes vrij, het record nummer en totaal aantal records. Tevens verschijnen systeem commando's in een window (zoals bij Archive in een window in het bovenste gedeelte). Onder de systeem commando's bevinden zich utilities voor file backup met overschrijving van bestaande files, het veranderen van de inhoud van een file, een directory variant met daarin opgenomen de file lengte, het samenvoegen van files, opsporen van tekst in de 'order' velden, het sturen van velden naar de printer, sturen van velden naar het beeldscherm en 'browsing' (bekijken) van records in aantallen van 20 tegelijk. Subfiles kan men selecteren tot maximaal 5 velden om verder te manipuleren, vanzelfsprekend kan ook selectie plaats vinden met numerieke argumenten.

Volgens de traditie van ARK is ook de beschrijving van het programma, een werkje van 10.000 woorden op A4 formaat, helder en duidelijk geschreven en heeft een logische opbouw. Voor de gebruiker prettig want de gewenste informatie is snel te vinden. Het boekje is gemaakt in de stijl van dat van de andere Archive tools van ARK, 'Cardfile', 'Mailfile' en 'Library file'.

Archivist was ontworpen met het oog op flexibiliteit en vervult daardoor een behoefte waar andere dergelijke programma's tekort schoten. Het was bedoeld voor de standaard 128k QL en werkt ook met microdrives (hoewel disk drive en geheugenuitbreiding praktisch gezien noodzakelijk zijn voor serieuze toepassingen - RWR).

De snelheid van het programma is te danken aan gebruik van machine code. Het is mogelijk om afdrukken te maken met Archivist, maar met een ander programma van ARK, 'Mailmerge', kan men bestandsgegevens exporteren uit Archive en een Quill \_lis file maken om geformatteerde documenten en brieven te vervaardigen.

Op deze manier is het zelfs mogelijk om Archivist files gemaakt op de QL te versturen naar bijvoorbeeld MS-DOS apparatuur en de ICL OPD ('One per desk', nooit gezien in Nederland - RWR).

Het programma is iedere cent waard, een krachtig database-tool en makkelijk en flexibel in gebruik.

Bron: QL World, march 1987

Auteur: Ron Massey

Vertaling: R. Reddingius, maart 87

### Toevoegen van velden in Archive.

In meer dan één Quasar en ook in QL-world hebben tips gestaan om aan een bestaand Archive-bestand velden toe te voegen. Bij de opbouw van een database is het een veelvuldig voorkomend verschijnsel, dat meer velden noodzakelijk zijn, dan oorspronkelijk bedacht. Als dit incidenteel voorkomt, is de in de QL-user-guide beschreven procedure een goede oplossing. Als je vaker met dit probleem worstelt, is het volgende Archive-programma wellicht een oplossing.

De procedure in de user guide heeft als belangrijk nadeel, dat je alle velden, dus ook de reeds bestaande in de procedure moet invoeren m.b.v. de Archive-editor en dus niet binnen een programma. De door mij bedachte procedure maakt gebruik van het SPOOLON- en LPRINT commando van Archive.

Met behulp van de functies NUMFLD(), FIELDN() en FIELDV() worden de velden van het bestaande bestand met LPRINT naar een procfil (ASCII) geschreven. Vervolgens kunnen de toe te voegen velden worden ingevoerd (worden ook met LPRINT weggeschreven). Het met SPOOLON gemaakte file in Archive "mergen" en aanroepen. Samen met de backup procedure wordt de bestaande file omgezet naar een nieuwe onder dezelfde naam. Het werkt altijd. Het bekende programma Dbasell kent deze functie standaard en werkt ook met het over en weer kopiëren van bestanden. Uiteraard kan het onderstaande programma ook omgebouwd worden zodat velden kunnen worden verwijderd en/of tussen-gevoegd; zover ben ik nog niet.

Voor mensen met een geheugenuitbreiding (op hun QL); deze procedures werken razendsnel via ramdisk!

### Het programma

Deze procedure maakt een backup van het bestand.

```
*proc bewaar
*print at 2,0;rept(" ",60)
* input at 2,0;"Welk bestand wil je wijzigen:
";fname$
* backup "mdv1_" + fname$ + "_dbf" as
"mdv1_oud_dbf"
* print at 20,0;rept(" ",80)
*endproc
```



Procedure voor een standaard schermheader.

```
*proc kop;x$
* print at 0,0; paper 2; rept(" ",80)
* print at 0,0; paper 2;"Datum:" ;date(1); at
  0,19;time()
* print at 0,29;x$; at
  0,63;int(memory()/1024);" kb vrij"
*endproc
```

Procedure om grote bestanden op 1 scherm te krijgen.

```
*proc page
* if regel=15: let regel=4: let
  kolom=kolom+27: endif
*endproc
```

Procedure om proc "crea" en "wissel" te maken.

```
*proc veld
* mode 0,8
* kop;"....."
* kill "mdv1_oud_dbf": kill
  "mdv1_nieuw_dbf"
* spoolon "mdv1_crea_prg" dump
* error bewaar
* if errnum()=100
* print at 20,0; paper 3;" Bestand
  "+fname$+" is geen databasefile. "
* bewaar
* endif
* kop;"Wijzig bestand """"+fname$+""""
* look "mdv1_oud_dbf" logical "o"
* let velden=numfld("o")
* let tel=0
* lprint "proc crea"
* lprint "create ""mdv1_nieuw" "logical ""n""""
* print at 2,0;"Het bestand bestaat uit de
  volgende velden:"
* print " "
* let regel=4: let kolom=0
* while tel<velden
* let veld$=fieldn(tel,"o")
* print at regel,kolom;tel+1;" " ;veld$
* lprint veld$
* let tel=tel+1
* let regel=regel+1
* page
* endwhile
* let ed$="j"
* print at 20,0; paper 3;" Geef ""n"" als je
  geen velden meer wilt toevoegen "
* while ed$="j"
* print at 18,0;"Nieuw veld:";
* input at 18,13;x$
* if lower(x$)<>"n"
* print at regel,kolom;tel+1;" " ;x$
* lprint x$
* let regel=regel+1: let tel=tel+1
* page
```

```
* else
* let ed$="n"
* endif
* endwhile
* lprint "endcreate"
* lprint "endproc"
* lprint "proc wissel"
* print at 20,0; rept(" ",80)
* input at 20,0;"Wil je deze wijziging bewaren
  (j/n)?";j$
* if lower(j$)="j"
* let nummer=0
* while nummer<velden
* lprint "letn."+fieldn(nummer,"o")+ "=o."+
  fieldn(nummer,"o")
* let nummer=nummer+1
* endwhile
* lprint "endproc"
* spooloff
* zetover
* else
* close "o"
* spooloff
* endif
*endproc
```

Procedure om gemaakte procs te mergen en uit te voeren.

```
*proc zetover
* print at 20,0; rept(" ",80)
* merge "mdv1_crea_prg"
* crea
* all "o"
* print at 20,0; paper 3;" "" ;recnum("o");" ";
* wissel
* append "n"
* endall
* print at 20,6; paper 3;" records gekopieerd "
* close "o": close "n"
* kill "mdv1_"+fname$+"_dbf"
* backup "mdv1_nieuw_dbf" as
  "mdv1_"+fname$+"_dbf"*endproc
```

NB de """" krijg je door in de editor "" te gebruiken.

Voorbeeld

Er is een bestand "joop" met de velden "adres\$" en "naam\$".

Je wilt het veld "telefoon\$" toevoegen.

De bovenstaande procedures maken dan de nu volgende procedures aan.

Crea maakt het "nieuwe" tijdelijke bestand aan.

```
*proc crea
* create "ram1_nieuw" logical "n"
* adres$
* naam$
* telefoon$
* endcreate
*endproc
```

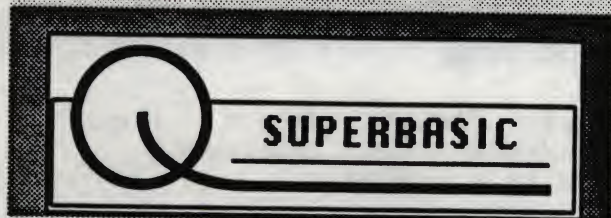
Wissel zet de velden uit het "oude" bestand over



op het nieuwe.

```
*proc wissel
* let n.adres$=o.adres$
* let n.naam$=o.naam$
*endproc
```

Jan Borst, Kerkelingsel 7,  
1981 BJ Velsen-Zuid  
tel. 02550-23294



QL-Talk

Uw QL floppy disks zodanig formatteren dat ze  
ook door de Atari ST kunnen worden gelezen en  
beschreven.

Hoewel de QL zonder floppy interface geleverd wordt, is door de ontwerper wel een 'floppy-formaat' standaard aangebracht. Hiermee zijn de floppies uitwisselbaar te maken, ook als de controllers van verschillende firma's zijn. Dat is eigenlijk niets bijzonders. Het succesnummer is een pseudofile, die vanuit SuperBasic bijzonder leuke dingen toelaat. Men opent een datafile

- \*Dxy -,

hierbij is x het aantal bytes per sector (0=128 Byte, 1=256 Byte, 2=512 Byte, 3=1024 Byte) en y de density (S=enkel (FM) en D=dubbel (FMF)).

Nu kan men met:

```
GET #kanaal\sector + zijde*156 +  
spoor*65536,a$
```

een sector volgens a\$ ingelezen worden. De functie QLSEC uit de listing staat u toe, met logische sector nummers te werken. Hiermee kan men zeer, zeer veel bijzondere formaten met de QL lezen. Het SANDY Superboard, de Pcml kaart, de Controller van Silicon Express, Cumana, en die van Kempston ondersteunen een directe toegang op de 3 1/2" Floppy-drive. De controller van de microdrives is met de Update van Qjump volledig Sinclair compatibel (de nieuwe versie is het al). In twijfel gevallen kunt u, voordat u koopt, dit eerst uit proberen. Het programma is getest met de Sandy kaart versie 1.17 & met de PCML kaart versie 1.14. Na inlezen van een monitor diskette en na het plaatsen van een Atari-ST floppy in de drive kunnen alle sectoren met - \*D2D - ingelezen worden.

De drive is één, het kopiëren van data een andere zaak. Het programma hiervoor is in Basic geschreven om het voor ieder begrijpelijk te maken.

Om een beter inzicht te krijgen wordt hier in het

### kort de organisatie verklaard van een Atari ST floppy.

Een Atari-floppy is in 80 sporen (0-79) ingedeeld. Elk spoor bezit 9 sectoren, daarmee zijn er dus totaal 720 sectoren.

Elke sector biedt plaats aan 512 bytes; op één zijde dus 360 KBytes. En dubbel zo groot voor dubbelzijdig. Daarbij is er verschil voor zijde 0 en 1.

Om nu exact een sector te benoemen zijn drie factoren nodig: 'zijde', 'spoor' en 'sector nummer'. Bovendien is er nog een logisch sector nummer, waarmee alle sectoren van 1 tot 1440 (of 720) genummerd worden. De logische sector 1 heeft betrekking op sector 1 in spoor 0 op zijde 0.

De formule luidt:

**logische sector = 9 \* (spoor \* aantal zijden + zijde) + sector.**

Afhankelijk van enkel of dubbelzijdige floppy wordt 1 of 2 ingevuld bij aantal zijden. De logische sector 1 speelt hier een belangrijke rol. Omdat zijn positie altijd éénduidig is, bevindt zich hier alle belangrijke informatie over de floppy zelf. Daarom wordt deze sector ook 'BOOTSECTOR' genoemd.

De floppy van de Atari ST is MS-DOS compatible, en omdat de eerste dertig bytes zeer belangrijk zijn, worden ze in de volgende tabel gegeven.

BYTE	INHOUD	BESCHRIJVING
1-2		Adres van een bootprogramma
2-11		versch. floppy kenmerken
12-13	\$0002	Byte per sector (hier 512)
14	\$02	Sector per cluster (hier 2)
15-16	\$0100	Gereserveerde sectoren bij begin van floppy (hier 1)
17	\$02	Aantal van FAT (catalogus sporen) hier 2
18-19	\$7000	Maximale hoeveelheid data in inhoud register (hier 112)
20-21	\$D002	Aantal sectoren op de floppy (hier 720=enkel zijdig) of
	\$A005	(dubbel zijdig)
	\$F8	Kenmerk voor enkel-zijdig of
	\$F9	Dubbelzijdig
23-24	\$0500	Aantal sectoren per FAT (hier 5)
25-26	\$0900	Sectoren per spoor (hier 9)
27-28	\$0100	Aantal zijden (hier 1) of
	\$0200	Dubbel zijdig
29-30	\$0000	Aantal verborgen sectoren (hier 0)

Daarmee kan uit sector nul alle informatie van een floppy gehaald worden. De PROCEDURE TESTX uit de listing benut deze informatie en de PROCEDURE make\_boot zal deze bij het



formatteren van een Atari St floppy op de QL samenstellen. In de tabel komt tweemaal het woord FAT voor. Deze afkorting betekent 'File Allocation Table'. Deze zijn 5 sectoren lang. Ze staan op de logische sectoren 2 tot 6 en 7 tot 11. Naar binnen volgt de inhoud van het floppy. Per file heeft de computer 32 bytes nodig. Omdat men maximaal 112 data namen in de inhoud kan bergen, legt men op 7 sectoren beslag ( $=112 \cdot 32/512$ ). Hiermee zijn de eerste sporen met de logische sectoren gevuld. De opbouw van een invoering van de inhoud ziet er als volgt uit:

#### BYTE Betekenis

1-8	Filenaam: Altijd HOOFDLETTERS
9-11	File extension Afkorting achter punt
12	File attribuut (zie tekst)
13-22	Niet gebruikt.
23-24	Tijd van verandering
25-26	Datum van verandering
27-28	Pointer op file begin (=eerste cluster)
29-32	File lengte in bytes

DE procedures "Attribute\$", "Datum\$", "Time\$", "Cluster" en "Lengte" decoderen deze informatie. Het Attribute-byte heeft de volgende betekenis:

Waarde	Betekenis
1	Read only file

2	Verborgen file, (verschijnt niet in Desktop)
4	\Systeem file
8	Disk naam
16	Overdracht is een Ordner
32	File is Archiv
64	Niet gebruikt
128	Niet gebruikt

De waarden zijn door optelling naar verkiezing te combineren. De waarde 7 is bijvoorbeeld, een read only verborgen systeem file. De datum en tijd is in gebruik als:

byte 24	byte 23	byte 26	byte 25
HHHHHMMM		MMMSSSSS	J
JJJJJJM	MMMDDDDD		

Hierbij moet men op het INTEL formaat letten. Bij de tijd is HHHHH het uur (0 - 23) MMMMMM de minuten (0 - 59) en SSSSS de seconden (0 - 59 in twee stappen) JJJJJJ het jaar (1980 - 2099), MMMM de maand (1 - 12) en DDDDD de dag (1 - 31).

De lengte is een "normaal" vier-byte-getal in Intel formaat.

(Volgende maand volgt het programma dat dit kunst en vliegwerk voor elkaar krijgt.®)

#### HEB MEELIJ!

De inzenders van dit stukje smaken het genoeg met Kees van der Wal onder één dak te wonen, en ook nog één gezamenlijk telefoonnummer te beheren. Alleen, zij zijn geen leden van jullie geachte vereniging. Wij werken overdag voor onze respektievelijke werkgevers ook met zulke apparaten, en zijn blij 's avonds en in de weekenden niet weer achter zo'n mono- dan wel polychrome beeldbuis te hoeven doorbrengen. Wij gunnen jullie je pleziertje uitermate, hoor, en Kees natuurlijk ook, maar wij maken andere keuzes als het gaat om het besteden van onze vrije tijd. Trouwens niet alleen onze vrije tijd is in het geding. Wij moeten ook nog eten koken, eten opeten en allerlei huishoudelijke klussen doen. Vervelend alleen dat wij daarbij nogal eens gestoord worden door leden van de QL-fanclub die vraagbaak Kees buiten de daarvoor geldende tijden menen te kunnen opbellen met vragen en problemen waar zij tegenaan gelo

pen zijn. Nu is het natuurlijk wel moeilijk, om, als je tussen 20.30 en 23.30 uur niet in staat bent op te bellen, je ongeduld te bedwingen, en natuurlijk is het heel verleidelijk om dan maar even buiten die uren te bellen; maar ... je bent niet de enige! en wij krijgen al die klanten hier op hetzelfde nummer aan de lijn. En moeten hem of haar weer verwijzen naar .... een tijdstip tussen 20.30 en 23.30! Mariet en John.



# PRISONER

## Het dilemma nader bekenen Henk ten Bos Rients Alberts

Als nieuw lid kreeg ik een aantal oude Quasars in handen en vond daarin ondermeer het artikel over het 'Prisoner's dilemma'. Nu ben ik, als informatica-student, altijd al geïnteresseerd geweest in problemen, vandaar dat ik eens wat dieper in de geleverde programma's dook en eveneens een medestudent het probleem voorlegde. We vonden het volgende:

-De oplossing van Diederik van Donk-(Quasar 23, pagina 418)

TABEL 1:

Een aantal malen Quasar - van Donk met bijbehorende puntenaantal. (Randomise bij van Donk weggelaten, anders steeds hetzelfde antwoord)

poging	pnt_a	pnt_b
1	171	176
2	153	158
3	250	250
4	195	200

De regelmaat in deze tabel frappeerde mij. Verdere bestudering van het programma leverde het antwoord:

Transactie 1 levert: actie\_a=1 en actie\_b=0 en bijgevolg A=0 en B=5. Vervolgens zijn er de volgende twee mogelijkheden:

1. Voor alle transacties geldt  $\text{rnd}(20) \leq 19$ , dan eindigt de cyclus in A=250 en B=250.
2. Voor zekere transactie n geldt:  $\text{RND}(20) > 19$ , dan vervolgt de cyclus met steeds actie\_a=0 AND actie\_b=0. Bijgevolg eindigt de cyclus in een toestand A=m en B=m+5 voor zekere m.

Het vermoeden was:

### Stelling:

De procedure van 'van Donk' is niet te verslaan! Men kan hoogstens een procedure schrijven die een gelijkspel bewerkstelligt.

### Bewijs:

De enige mogelijkheid om beter te scoren dan van Donk is, wanneer actie\_ik=0 en actie\_Donk=1. Dan scoor ik namelijk 5 punten en van Donk niets. Wanneer kan dit optreden? Bij de eerste transactie niet, want dan kiest van Donk actie=0. Het moet dus in een latere actie gebeuren, nl. wanneer van Donk voor de eerste keer actie\_Donk=1 neemt. Op dat moment kiezen wij actie\_ik=0 en scoren.

Echter, wanneer wordt actie\_Donk voor de eerste keer 1?!?

Uit het programma blijkt: als de tegenspeler actie\_ik=1 in de vorige poging genomen heeft. In de vorige poging was actie\_Donk echter 0 (we hadden verondersteld dat dit de eerste keer actie\_Donk=1 was!) oftewel in de vorige poging heeft van Donk 5 punten gescoord en wij niets. Met onze winst komen wij dus slechts op gelijke hoogte en kunnen zo nooit winnen. De procedure van 'van Donk' is nog eenvoudiger te maken:

```
PROCEDURE speler_a actie_a=0
END DEFINITION speler_a
```

Speel gewoon iedere keer vals. Speelt de tegenstander ook vals, dan levert dat ieder 1 punt op. Is de tegenstander een keer zo dom om eerlijk te handelen, dan is de winst: 5 punten voor mij. Het is dus weinig zinvol om te proberen een programma te maken om van Donk te verslaan, want dit lukt niet. Daarom is het misschien wel leuk om het probleem te verleggen naar het handelsdilemma:

### Opgave:

Schrijf een procedure die tegen de al aanwezige procedures een zo groot mogelijke winst behaalt. Hieruit blijkt al dat de procedure van D. van Donk nooit een grote winst kan halen tegen zichzelf.

### Voorbeelden:

Een 0-en genererende procedure is onverslaanbaar en zal per transactie meestal 1-1 spelen. Dit is dus een matige winst (100 na 100 transacties).

Een 1-en genererende procedure kan geen enkele andere procedure verslaan, maar zal tegen zijn tegenstander toch meestal 3-3 spelen. Dit levert dus na 100 transacties een grote winst (300): een betere procedure.

### Conclusie:

onderstaande procedure zal waarschijnlijk van geen enkele procedure winnen, maar na tegen meerdere tegenstanders zaken te hebben gedaan zal deze toch een van de procedures zijn die grote (de grootste?) winst heeft geboekt!

```
1000 DEFINITION PROCEDURE speler_a
1010 REM R. Alberts, Groningen
1020 IF transaktienummer=1 THEN
1030 transactie_a=1
1040 RETURN
1050 END IF
1060 IF vorige_b=1 THEN t=t+1
1070 IF transaktienummer<10 THEN
1080 transactie_a=1
1090 ELSE
1100 IF t/transaktienummer<.5 THEN
1110 transactie_a=0
1120 ELSE transactie_a=1
1130 END IF
1140 END IF
1150 END DEFINITION speler_a
```



# OVER DATABASES EN ARCHIVE... Bob Visser.(1)

## deel 1

Op p.403 en volgend van Quasar doet Ron een m.i. geslaagde poging gegevensbeheer in het algemeen en het gebruik van Archive in het bijzonder onder Uw alier aandacht te brengen. Omdat ik geconstateerd heb dat er nogal wat misverstand heerst over begrippen als "electronische kaartenbak", "databases", "relationeel" e.d. hierbij een poging mijnerzijds een en ander toe te lichten. We beginnen eenvoudig.

Neem een kaartenbak. Hierin zitten kaarten, met op die kaarten allerlei gegevens. Voor iedere herkenbare eenheid in de werkelijkheid (een boek, een persoon, een order e.d.) houden we een kaart bij.

We hebben:

1. gegevenselementen, ofwel attributen, velden, variabelen e.d.
2. herkenbare eenheden of objecten. Zo'n eenheid kan zijn: een persoon.

Aan het OBJECT 'persoon' kunnen we diverse ATTRIBUTEN onderscheiden: armen, benen, hoofd e.d. Voor iedere persoon kunnen we een 'kaart' aanleggen in onze kaartenbak. Overigens: 'persoon' noemen we vaak het OBJECTTYPE; hiermee bedoelen we de min of meer abstracte verzameling personen. 'Jansen' is een OCCURENCE van dit objecttype.

In meer gewone termen: een objecttype wordt vastgelegd met behulp van een BESTAND, een occurrence met behulp van een RECORD. De kaarten in een kaartenbak hebben meestal een bepaalde volgorde. Deze volgorde wordt bepaald door het SLEUTELATTRIBUUT. Meestal vinden we dit het belangrijkste gegeven, bijv. de naam van een persoon. Het zoeken op het sleutelattribuut in de kaartenbak gaat meestal lekker snel; we kunnen sprongsgewijs zoeken (vergelijk zoeken in het telefoonboek). Willen we op een ander attribuut gaan zoeken (bijv. op het voorkomen van bepaalde straatnummers in het telefoonboek), dan moeten we ieder object afzonderlijk bekijken. Dat duurt dus wel even. In de praktijk zien we dan ook dat er vaak meerdere 'kaartenbakken' of indexlijsten worden bijgehouden over dezelfde objecten, zodat via meerdere ingangen de gegevens geraadpleegd kunnen worden.

Dus: een kaartenbak met al het personeel gesorteerd op naam, een andere bak gesorteerd op geboortedatum, een aantal lijsten met het personeel gesorteerd naar de afdelingen waartoe zij behoren e.d. Het bijhouden van diverse kaartenbakken (of administraties) over dezelfde objecten is een tijdrovende en vooral lastige klus. Want o wee als je vergeet bepaalde mutaties in alle

daartos bestemde administraties door te voeren!

We noemen dit het gevaar van INCONSISTENTE gegevens: de data spreken elkaar tegen. In een beetje omvangrijk bedrijf heb je vele verschillende gegevensverzamelingen. Over het algemeen heeft iedere afdeling z'n eigen administraties. Gegevens over klanten kunnen aanwezig zijn bij de afdeling Verkoop, de afdeling Facturering, de afdeling Marketing e.d. Via een ingewikkeld stelsel voorschriften en veel kleurige bonnetjes etc. wordt getracht deze gegevens consistent te houden. Dat lukt vaak maar zeer gedeeltelijk. Tot zover de niet-geautomatiseerde situatie.

Nu komt ons gewaardeerd Sin\_QL\_Air lid op een idee: als ik nu eens de kaartenbak waarvoor ik verantwoordelijk ben in m'n QL stop, dan hoef ik al die neven-lijsten ook niet meer bij te houden: de computer neemt het tijdrovende zoeken voor zijn rekening! Hij probeert het eerst eens thuis uit met een bestandje van z'n... boekenverzameling.

Even nadenken: een boek is hier dus het OBJECT; aan dit object kunnen ATTRIBUTEN onderscheiden worden. Welke? Dat maakt de gebruiker ukeraard zelf uit, want dat is geheel afhankelijk van de doelstelling van de administratie. Wat wil ons QL-lid? Ten eerste zijn boekenbezit beschrijven, ten tweede bepaalde boeken snel op kunnen zoeken en ten derde het uitleenen van boeken bijhouden. Na enig gedub komt hij tot de volgende relevante eigenschappen: auteur, titel, jaar van uitgave, taal, onderwerp, vindplaats, gelezen?, uitgeleend aan, uitgeleend op.

Mooi, snel achter de zwarte toetsen en ARCHIVE openen. Daar gaan we:

```
CREATE
boeken      auteurs$  titel$   jaar   taal$
onderwerp$  vindplaats gelezen$
uitgeleend$ uitgeleend$ ENDCREATE
```

In een andere (meer volwassen) kaartenbakprogramma (DBMS) zou je meer keuzes hebben t.a.v. het FORMAAT van je attributen, en zou je ook op moeten geven hoeveel tekens een attribuut maximaal kan bevatten, bijvoorbeeld:

auteur	TEKEN	30
titel	TEKEN	60
jaar	CIJFER	4
taal	TEKEN	15
onderwerp	TEKEN	60
vindplaats	CIJFER	6
gelezen	BOOLEAN	1
uitgeleend	TEKEN	15
uitgeleend op	DATUM	6



ARCHIVE kent dus maar twee formats, TEKEN (door een \$ achter de veldnaam te zetten) en CIJFER (door geen \$ achter de veldnaam te zetten). De meeste andere DBMS'en kennen minimaal de hiervoor genoemde vier data-types. Wat hebben we nu gedaan? We hebben het raamwerk gedefinieerd voor ons BESTAND van boeken. Met behulp van de computer wordt hier een FILE van gemaakt. We hadden echter net zo goed een kaartenbak en een pak fiches kunnen kopen: het principe is precies hetzelfde.

We spreken voor dit soort toepassingen dan ook van een ELECTRONISCHE KAARTENBAK. Ons bestand heeft in principe precies dezelfde mogelijkheden als een 'echte' kaartenbak!!

Het enige verschil is dat we in ons computer-bestand niet zelf hoeven te zoeken: dat doet de computer voor ons. Let op: een elektronische kaartenbak bevat dus gegevens over één bepaalde verzameling OBJECTEN, in dit geval boeken. Pas als we over verschillende soorten objecten gegevens gaan bijhouden spreken we over een DATABASE! In een database worden dus een (groot) aantal kaartenbakken geïntegreerd. Dit betekent dat een database ook een (groot) aantal bestanden (files) omvat, terwijl een elektronische kaartenbak altijd maar uit één bestand bestaat. Levert een elektronische kaartenbak eigenlijk geen enkel extra voordeel boven een handmatige administratie, een database doet dat wel. Volgende keer.... meer!

« Bob Visser »

**Als u niet meer weet of u al betaald heeft, kijk dan op uw etiket.**

**f40 betekent dat u nog moet betalen.**

**f40 'piek' voor een 'piekfijn' blad!**

#### Lijst van Verkrijgbare onderdelen.

Keyboard	bovenkant (compleet) (gebruikt)	
Bubbelmat		15.-
Membraan		17.50
Losse toetsen (gebruikt)		
Microdrive unit		60.-
ULA Microdrive 2G-007(IC29-IC30)		25.-
MC 68008 (IC18)		105.-
8049 QL (IC24)		30.-
ZX8302 (IC23)		50.-
ZX8301:CLA2310 (IC22)		40.-
HAL 8449QL (IC38)		40.-
JS Eproms compleet		50.-
JM Eproms compleet (gebruikt)		
AH Eprom 8000 (gebruikt)		

Prijzen zijn inclusief verzendkosten.

GEHEUGEN-UITBREIDING 512 K	220.-
DISC-CONTROLLER+RAMDISK	275.-

GEVRAAGD (VOOR DE CLUB IVM.DATABANK):

EEN QL (mag eventueel defect zijn)

Te koop: Brother EP22. Prijs n.o.t.k.

Kopij alleen op cartridges opsturen.  
Het formaat is voorlopig nog niet echt belangrijk.

SLUITINGSDATA INZENDEN KOPIJ:

Quasar 26: 7 mei  
Quasar 27: 4 juni  
Quasar 28: 9 juli  
Quasar 29: 27 augustus

Het 500ste lid krijgt, zoals beloofd, een  
Miracle modem kado!  
De overhandiging hiervan zal geschieden  
op de bijeenkomst van 23 mei in Utrecht.

De gelukkige is de heer E.Hoogcarspel  
uit Schiedam.

Gefeliciteerd.



# Een weerfoto van METEOSAT.

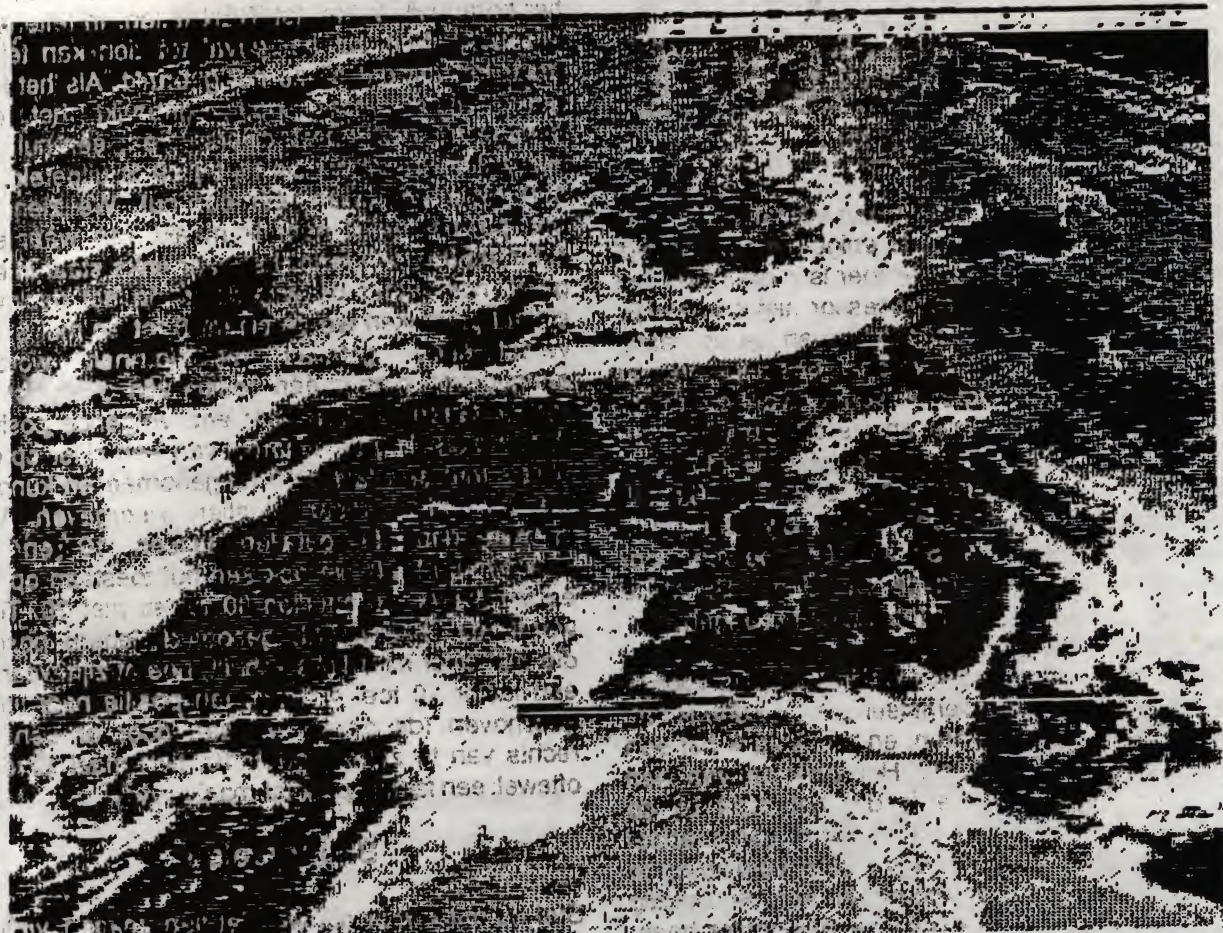
(33000 km hoogte)

Gedecodeerd en gesprint mbv een OL

Het programma hiervoor is binnenkort beschikbaar

Op zaterdag 23 mei is er een bijeenkomst in Utrecht. Adres: Grootbegijnlaan 15. School: Ir AJ Versfelt MTS.

Op deze bijeenkomst willen we proberen een 'spontaan' vraag en antwoord-uurtje te houden.



print

*Handwritten signature or scribble.*



\_\_\_\_\_

Omdat het niet handig is om steeds met de wagen van de typemachine te knoeien is er tenslotte nog een derde mogelijkheid om het grondtal aan te geven; men plaatst een teken vóór het getal. Voor



# **Machinetaal**

## **cursus**

# **Hoofdstuk 1**

## **Inleiding**

Deze cursus valt uiteen in een aantal delen. Ten eerste is het vereist dat U enige kennis verkrijgt van het binaire rekenen. Verder is kennis over het inwendige van de microprocessor niet ontbeerlijk, terwijl ook de nodige aandacht aan de operating software zal moeten worden besteed. Verder dient U op de hoogte te komen van de terminologie. Een aantal boeken die aansluiten op en bij deze cursus is 'QL Advanced User Guide' door Adrian Dickens (uitgeverij ADDER), 'The Sinclair QDOS companion' en 'Assembly language programming on the Sinclair QL' beiden door Andrew Pennell, uitgegeven door Sunshine uitgeverij. Alle boeken zijn (helaas) in het engels.

Veel mensen vinden het moeilijk zich iets voor te stellen bij een stuk assembleertaal: hoe doe ik het, hoe geef ik de opdrachten, en hoe krijg ik het een en ander aan de praat? Het principe van een machinetaal programma is dat het dicht bij de machine staat. Zo dicht echter, dat er een speciaal programma nodig is om ons niet in verwarring te brengen. We programmeren in een eenvoudig taaltje. Dit 'gooien' we door de assembler heen. Deze vertaalt het programma dat we hebben opgegeven naar enen en nullen, zodat de processor ze kan uitvoeren. Dit uitvoeren gaat echter zo snel dat we wederom een hulpprogramma nodig hebben om te kunnen zien wat er gebeurt. Een 68008 kan vele instructies per seconde uitvoeren. Een monitor is heel verwarrend, ik geef het toe, is een programma dat ons in staat stelt stap voor stap, of met beheersbare sprongetjes door ons programma heen te lopen. De programma's en hun gebruik komen later pas aan de orde. Eerst moeten we ons wat vertrouwd maken met de nodige theorie. Later kunnen we een aantal sessies uitvoeren waarin we de programma's gebruiken en het eigenlijke programmeren doen. Misschien is dit het uitstek een item om op bijeenkomsten te organiseren. Eerst dus droogzwemmen.

### **Binaire tellen**

### **Positie stelsel**

Een kind dat leert rekenen begint eenvoudig. Tellen is de eerste stap: één, twee, drie en verder,

tot 9. Nu heeft de natuur ons uitgerust met tien vingers, dus op een gegeven moment is de kleine toe aan 10. Vaak meldt hij of zij dan aan ieder die het horen wil: ik kan tot TIEN tellen. In feite is dit bijzonder. Niet dat het geval tot tien kan tellen, maar het feit dat hij de nul gebruikt. Als het kind geen cijfers meer weet, gebruikt het een combinatie van lagere cijfers. Tien; één-nul, 10. Het zelfde treedt op als hij bij 99 is uitgeteld, en 100 moet gebruiken. één-één-nul. We zien dat ons positie-stelsel gebruik maakt van waar een cijfer staat. Hoe verder het naar links staat in een getal, hoe groter de waarde is. In het decimale stelsel gebruiken we de cijfers 0 tot en met 9. De verschillende getallen die kunnen worden aangegeven binnen één positie zijn 0 tot en met 9. Gaan we dan naar een volgende positie, dan is plotseling de waarde van een cijfer op die plaats met een factor 10 toegenomen en kunnen we bovendien meer getallen weergeven. We kunnen dan de getallen tussen 10 en 99 aangeven, of, als we ook een nul toestaan op de eerste plaats, de getallen 00 tot en met 99. Tien wordt het GROND TAL genoemd. Iedere positie die een cijfer naar links schuift, neemt zijn waarde een factor 10 toe. 1 (één) een positie naar links geschoven (dit geven we aan door er een '0' rechts van te zetten) geeft 10, tiert maal zoveel, oftewel: een factor 'grondtal' maal zoveel.

### **Binaire stelsel**

Stel nu eens voor dat we niet tien, maar 1 vinger gehad zouden hebben. Behalve dat het knap lastig eten is, zouden we ook minder goed kunnen tellen in het tientallig stelsel. Immers, tot één tellen gaat nog wel, maar dan komen we al snel in de moeilijkheden, want twee kennen we helemaal niet. We zouden dan dus al snel het kind horen zeggen: "ik kan tot twee tellen". Passen we namelijk de truc van het kind toe dan kunnen we wel tot twee tellen: één, één-nul. Onmiddellijk gaan we hierbij dus over naar de volgende positie. Daar is de '1' die we neerzetten een factor GROND TAL toegenomen. GROND TAL was nu net het getal dat we niet konden vormen binnen de ons beschikbare cijfers. Blijkbaar is hier dus het grondtal 2. Noemden we het tientallig stelsel DECIMAAL, het tweentallig stelsel met grondtal 2 noemen we BINAIR. We kunnen dus nu in het binaire stelsel tellen tot drie: 'nul' wordt weergegeven door 0 ( of 00 als we twee cijfers gebruiken), en we tellen verder: 01, 10, 11. Dan zijn we dus tot drie gekomen. Nu moeten we verder met vier. Hiervoor moeten we een volgende positie de derde gebruiken, waarbij een cijfer dat op deze positie staat, de waarde  $2^2=4$  ( $2^2$  is 2 tot de tweede macht is  $2 \times 2=4$ )